Topological trajectory classification with filtrations of simplicial complexes and persistent homology

Journal name
000(00):1–13

DOI:doi number
http://mms.sagepub.com

Topological Trajectory Classification with Filtrations of Simplicial Complexes and Persistent Homology

Florian T. Pokorny[*], Majd Hawasly[†] and Subramanian Ramamoorthy[†]

Abstract

In this work, we present a sampling-based approach to trajectory classification which enables automated high-level reasoning about topological classes of trajectories. Our approach is applicable to general configuration spaces and relies only on the availability of collision free samples. Unlike previous sampling-based approaches in robotics which use graphs to capture information about the path-connectedness of a configuration space, we construct a multiscale approximation of neighborhoods of the collision free configurations based on filtrations of simplicial complexes. Our approach thereby extracts additional homological information which is essential for a topological trajectory classification. We propose a multiscale classification algorithm for trajectories in configuration spaces of arbitrary dimension and for sets of trajectories starting and ending in two fixed points. Using a cone construction, we then generalize this approach to classify sets of trajectories even when trajectory start and end points are allowed to vary in path-connected subsets. We furthermore show how an augmented filtration of simplicial complexes based on an arbitrary function on the configuration space, such as a costmap, can be defined to incorporate additional constraints. We present an evaluation of our approach in 2, 3, 4 and 6 dimensional configuration spaces in simulation and in real-world experiments using a Baxter robot and motion capture data[1].

[*] Centre for Autonomous Systems, CAS/CVAP,
KTH Royal Institute of Technology, Sweden,
fpokorny@kth.se
[†] IPAB, School of Informatics, University of Edinburgh, UK
[1] Parts of this work were presented in (Pokorny et al., 2014) at Robotics: Science and Systems 2014. This work contains an extended discussion, theory section and experimental evaluation.

1. Introduction

For robots to autonomously operate in a wide variety of environments, we require algorithms and data structures that enable such systems to reason rigorously about the collision-free subset $\mathcal{C}_f$ of their configuration space. Furthermore, it is desirable for such systems to be able to classify continuous trajectories in their configuration space in order to reason about the space of possible motions.

In the case where $\mathcal{C}_f$ is explicitly describable in a noise-free manner in terms of semi-algebraic functions, analytic methods (Canny, 1988; Latombe, 1991; Schwartz and Sharir, 1983) provide an elegant classical avenue to studying $\mathcal{C}_f$. With the increase in computational resources and the availability of large data sets, the *data-driven paradigm* has more recently provided an alternative to analytical methods; providing a more powerful and usable solution in many scenarios where analytic models are either infeasible or computationally too expensive to obtain. An example of this approach is the use of Gaussian Mixture Models in the Programming by Demonstration framework (Billard et al., 2008), where trajectories are modelled as integral curves of vector fields obtained from a trained Gaussian Mixture Model. Similarly, in motion planning, an explicit description of $\mathcal{C}_f \subset \mathbb{R}^d$ is often not available and popular algorithms such as Rapidly-exploring Random Trees (RRT) and Probabilistic Roadmaps (PRM), (Kavraki et al., 1996; LaValle, 2006; LaValle and Kuffner, 2001), are based on the idea of utilizing a set of random samples $X \subset \mathcal{C}_f$ to construct a graph Γ with vertices in $\mathcal{C}_f$ and where edges correspond to local paths which can be determined by a local path planner. The graph Γ can then be used to efficiently carry out

motion planning. When $\mathcal{C}_f$ is a tame space, Γ, for sufficiently large X, provides an approximation of $\mathcal{C}_f$ which allows us to answer basic questions about the path-connectivity of $\mathcal{C}_f$.

We observe that both, a large class of current probabilistic approaches such as Gaussian Mixture Models and Gaussian Processes (Deisenroth et al., 2013; Rasmussen and Williams, 2006), as well as graph-based approaches in motion planning are currently not able to make use of *homological information* about $\mathcal{C}_f$. In particular, these methods are not able to detect whether trajectories are homotopy inequivalent – *i.e.* pairs of trajectories that cannot be continuously deformed into each other. The reason for this is that, besides approximating the path-connectedness of $\mathcal{C}_f$, unaugmented RRT and PRM graphs do not store information about higher order topological information contained in higher homology and homotopy groups and, similarly, probabilistic methods have so far not incorporated such information.

In this paper, we use filtrations $\mathcal{F} = \{\mathcal{F}_r : r \geqslant 0\}$ of simplicial complexes defined in terms of random samples $X \subset \mathcal{C}_f$ as a key technique for representing configuration spaces in practical robotics applications. Such filtrations are common in topological data analysis (Carlsson, 2009), but have to the best of our knowledge not been utilized to model multi-joint robot configuration spaces prior to our work (Pokorny et al., 2014). We in particular propose a novel approach to trajectory classification based on filtrations of simplicial complexes defined in terms of random samples $X \subset \mathcal{C}_f$. From such filtrations, we then extract higher-order topological information for the purpose of understanding and classifying equivalence classes of trajectories in $\mathcal{C}_f$. Given a sufficiently good approximation of $\mathcal{C}_f$ by $\mathcal{F}_r$, our approach yields a finite set of equivalence classes of trajectories with the property that *no trajectory belonging to one equivalence class can be continuously deformed to any trajectory in any of the other equivalence classes*. Note that there is a subtle difference to an exact classification by homotopy classes in the statement above. Our classification is coarser in the following precise sense: we use the first homology group for classification, which yields a weaker classification than a classification by the first homotopy group. In particular, there exist cases where two trajectories which are considered equivalent in homology are not homotopy equivalent, but two trajectories in different homology classes will always be homotopy inequivalent. The main benefit of working with homology is that homology groups are purely linear algebraic objects that can be computed efficiently while homotopy groups are generally complicated noncommutative groups. Our filtration $\mathcal{F}$ is based on Delaunay-Čech complexes which depend on a scale parameter r and which have recently been proven by Bauer and Edelsbrunner (2014) to provide a homotopy-equivalent reconstruction of the space $X_r = \bigcup_{x \in X} \mathbb{B}_r(x)$, where $\mathbb{B}_r(x) = \{y \in \mathbb{R}^d : \|x - y\| \leqslant r\}$. Our work utilizes persistent homology as described in Carlsson (2009); Edelsbrunner and Harer (2008, 2010) which generalizes classical homology groups to a multiscale setting – meaning that we are able to compute topological information about the analogue $\mathcal{F}_r$ of Γ for all scales $r \geqslant 0$ simultaneously without having to choose a particular scale upfront.

Additionally, the 1-skeleton $\mathcal{F}_r^1 \subseteq \mathcal{F}_r$ is a graph which can be used for path-planning. To summarize, our main contributions are the following:

- We introduce and evaluate the use of *filtrations of simplicial complexes* as well as *persistent homology* for modelling robot configuration spaces and sampled trajectory data arising in robotics.
- We propose an algorithm based on persistent homology to classify trajectories with fixed start and end points topologically. We then show how, using a cone construction, our approach can classify trajectories whose start and end-points lie in connected sets S, T respectively.
- We show how to use our approach when a general filtration function is considered, and we provide an example of filtrations arising from a cost-function.
- We provide an experimental evaluation of our approach in configuration spaces of dimension 2, 3, 4 and 6 using Delaunay-Čech complexes both in simulation, with motion capture data of a human-robot interaction, and with trajectories recorded using a Baxter robot.

The paper is structured as follows: In Sec. 2 we discuss the motivation of our approach and the differences and similarities with prior works. Sec. 3 reviews the theoretical background on persistent homology and filtrations. We discuss our approach to trajectory classification in Sec. 4 and present an experimental evaluation in Sec. 5. Finally, we conclude with Sec. 6 and discuss future work.

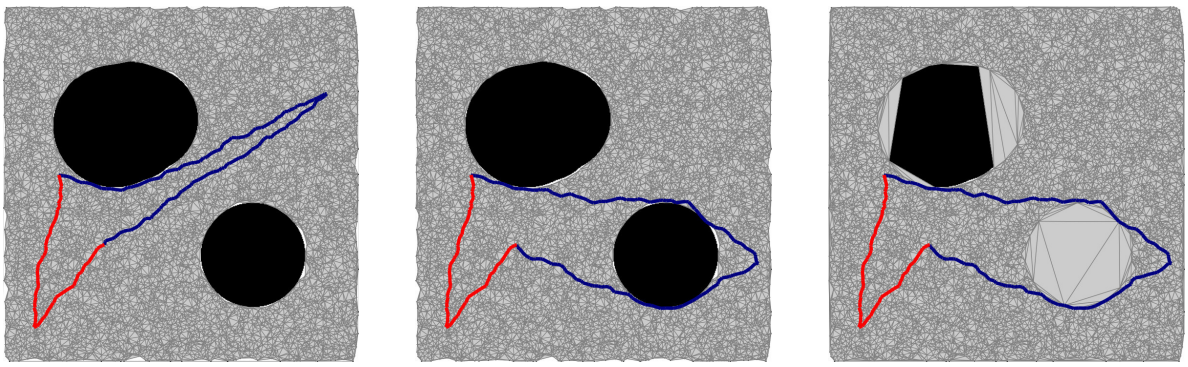

Fig. 1. We display a rectangular configuration space $\mathcal{C}$ of side-length 500 and with two obstacles (in black). Several trajectories are depicted in red and blue. The gray area displays an approximation of $\mathcal{C}_f$ by a Delaunay-Čech complex $DC_r(X)$ from 10000 samples $X \subset \mathcal{C}_f$ and with filtration value $r = 11.07$ (left and middle figure) and $r = 73.76$ (right figure). All of our figures are best viewed in color.

2. Motivation and Related Work

For a robot to reason efficiently about trajectories within its own free configuration space $\mathcal{C}_f$, or about the motions of other human or robot agents in its environment, a suitable partitioning of continuously varying families of trajectories into a discrete set of equivalence classes is desirable.

Clustering trajectories is difficult in general since trajectories can have varying length and are not immediately representable as vectors in a vector space of fixed dimension as required by commonly used algorithms such as support vector machines (Cristianini and Shawe-Taylor, 2000).

Several approaches to the classification of trajectories, as reviewed in (Zheng and Zhou, 2011), are based on various approaches to measuring the dissimilarity between trajectories, such as the Hausdorff distance, edit distances and dynamic time warping. For the purpose of activity analysis, the work of Morris and Trivedi (2009) reviews trajectory clustering approaches based on various clustering algorithms and distance measures. These methods for clustering are related to other approaches for simplifying trajectories by taking into account topological attributes. For instance, Katsikouli et al. (2014) present an approach to eliminate spurious features in trajectories, such as GPS traces of human mobility data, in order to reduce the burden on further analysis.

In robotics, the knowledge of classes of trajectories is beneficial for example in the programming by demonstration framework (Billard et al., 2008) where movement primitives of a robot's behavior are constructed from initial trajectory demonstrations provided by a human teacher. Equivalence classes of robot trajectories can furthermore be useful in order to reason about alternative trajectories when a subset of trajectories becomes invalid due to changing environment conditions. The recent work of Knepper et al. (2012) has demonstrated the usefulness of trajectory classes of local paths to improve the efficiency of a motion planning algorithm in particular. Another reason to adopt an approach such as ours is the need to reason about task level attributes of robot motion, abstracting over, e.g., the variability implicit in the many ways a robot could traverse between two regions in the configuration space. In Sisbot and Alami (2012), complex human-robot manipulation tasks are made tractable by reasoning at the level of object hand-over points and task segments, for example. Our experiments, in section 5, illustrate how our classification procedure can be used to extract such qualitative movement primitives.

Purely topological approaches to the analysis of trajectories in $\mathcal{C}_f$ focus on notions of equivalence which do not depend on a metric. There, two paths $\alpha, \beta : [0, 1] \to \mathcal{C}_f$ with $\alpha(0) = \beta(0)$, $\alpha(1) = \beta(1)$ are called homotopy equivalent if α can continuously be deformed to β in $\mathcal{C}_f$ while keeping the end-points fixed. We use Fig. 1 as a running example for illustration. Note that all figures in this paper are best viewed in color. For now, consider $\mathcal{C}_f$ as being approximated by the gray region, while the black regions correspond to obstacles. Several trajectories with identical start and end-points are depicted in red and blue. The blue trajectory in the leftmost figure is homotopy equivalent to the red trajectory, while the trajectories in the middle figure are not homotopy equivalent. Note that, while trajectories can have identical

distance in $\mathbb{R}^2$ under *e.g.* the Hausdorff distance, they may or may not be homotopy equivalent. When the lower obstacle is removed in the right figure, the red and blue paths from the middle figure become homotopy equivalent, for example. While homotopies between selected trajectories can in some cases be constructed explicitly, for example using a curve shortening algorithm (Chou and Zhu, 2001), this approach can fail in several cases. Firstly, an explicit homotopy construction, can only establish the existence of a homotopy in some cases, but may fail, for example when there exist just a single but multiple geodesics between two points. Furthermore, numerical stability and resolution pose challenging problems for such methods, and the failure to construct a homotopy between two curves does not imply homotopy inequivalence. Our approach instead *does allow us to prove the homotopy inequivalence of trajectories* when the underlying configuration space is sufficiently well approximated by a simplicial complex.

Topological concepts such as retractions and cell decompositions have played a key role in classical approaches to motion planning (Latombe, 1991). There, $\mathcal{C}_f$ is typically assumed to have a known algebraic or semi-algebraic structure. The visibility graph in 2D and retraction-based methods rely on constructing a graph using which motion planning is performed. The general roadmap method of Canny (1988) uses ideas closely related to Morse theory and projections to lower-dimensional spaces to obtain a complete motion planner for semi-algebraic sets. Similarly, the seminal work of Schwartz and Sharir (1983) proceeds by constructing an exact cell decomposition by means of a cylindrical algebraic decomposition of $\mathcal{C}_f$. This is related to our approach since our simplicial complexes form a particular type of approximate cell decomposition. In Schwartz and Sharir (1983), homology groups of $\mathcal{C}_f$ are computed by an exact cell decomposition $\mathcal{Z}$ and the general path planning problem is solved using $\mathcal{Z}$. These classical works have to the best of our knowledge however not considered the use of the first homology group of $\mathcal{C}_f$ for trajectory classification, and the focus has been on motion planning and not classification. An important difference to our work is the fact that we only assume the knowledge of *potentially noisy point-samples* from $\mathcal{C}_f$ using which we build a *simplicial complex filtration* rather than assuming a known description of $\mathcal{C}_f$ as a (semi-)algebraic set. Furthermore, our approach allows us to study the homotopy equivalence of paths within the neighborhood X_r of a set of samples $X \subset \mathcal{C}_f$ more generally, *e.g.* when X_r does not yield a reconstruction of the full space $\mathcal{C}_f$. In a more recent related work, Zhang et al. (2007) construct an approximate cell decomposition using a recursively refined decomposition of $\mathcal{C}_f$ into hypercubes to ensure a sufficiently fine reconstruction of $\mathcal{C}_f$. However, only the path-connectivity of this decomposition is then used for motion planning and homological properties are not further investigated.

Our work is also related to sampling-based algorithms constructing a graph Γ from X to answer questions about the path-connectivity of $\mathcal{C}_f$. RRTs and PRMs (Kavraki et al., 1996; LaValle, 2006; LaValle and Kuffner, 2001), in particular, are examples of these which have attracted unabated interest since their invention (Karaman and Frazzoli, 2011; Lindemann and LaValle, 2005; Masehian and Sedighizadeh, 2007). The graph Γ can be thought of as an approximation of $\mathcal{C}_f$ from X. The filtrations of simplicial complexes used in our work extend the concept of a graph to a multi-scale approach which can recover more detailed information about $\mathcal{C}_f$. Such filtrations depending on a scale parameter r have been used in topological data analysis (TDA) (Carlsson, 2009; Edelsbrunner and Harer, 2008, 2010) to study the persistent homology groups which we use here and which capture information about the topology of data at all scales simultaneously. The origins of practical multi-scale simplicial complex filtrations trace back to early work by Edelsbrunner (1992); Edelsbrunner and Mücke (1994) on alpha-shapes and the introduction of persistence in (Edelsbrunner et al., 2002). In the field of sensor networks, persistence has also been utilized to solve multi-sensor coverage problems (de Silva and Ghrist, 2006, 2007). In machine learning, persistent homology has been investigated as a tool for topology aware kernel density estimation, topological feature extraction and inference (Chazal et al., 2013; Fasy et al., 2013; Pokorny et al., 2012).

One of the advantages of the knowledge of homotopy classes is that a motion planning algorithm can utilize efficient replanning within each such class (Brock and Khatib, 2000). Since local variational or gradient based methods can continuously deform trajectories towards local optimality only within the homotopy class of an initial trajectory, it is advantageous to maintain a set of homotopy inequivalent

trajectories each of which can then be optimized using these methods. Topological information about path classes hence allows us to incorporate non-trivial *global* information with these *local* methods.

Recent approaches which attempt to obtain equivalence classes of paths include the works of Jaillet and Siméon (2008a,b) on path deformation roadmaps where a graph-based representation to plan in the space of paths up to a class of continuous deformations is proposed. Recently, researchers have in particular investigated homotopy classes of trajectories in explicitly described spaces. Using the residue theorem of complex analysis (Bhattacharya et al., 2010) studied an application of *homology classes* to motion planning in 2D in the case where the obstacles in $\mathcal{C}$ can be contracted into representative and explicitly defined points. In Bhattacharya et al. (2011), this was extended using electromagnetism theory and Ampère's law to the case of 3D. There, obstacles were assumed to be contracted into skeletons and then modeled as current-carrying wires. Similarly to our work, the authors argue that homological information is useful and computationally favorable to more general homotopy invariants in robotics. In Bhattacharya et al. (2012), a generalization to arbitrary dimension is proposed and an integration of differential 1-forms over cycles is shown to be sufficient to determine topological classes using the language of de Rham cohomology theory. In Kim et al. (2012), motion planning in 2D with homology constraints is formulated as a mixed-integer quadratic program by endowing path segments with binary labels that identify their relation to the domain obstacles. A problem the above recent approaches suffer from is that *they require an explicit description of the obstacles in the configuration space*, e.g. in 2D as a union of shapes each of which is contractible to a geometrically specified point $p \in \mathcal{C} - \mathcal{C}_f$. In many cases, such information is however not easily available for real robotic systems or too expensive to compute. We instead propose a *data-driven, sampling-based approach* to building a representation of $\mathcal{C}_f$ from which topological information about trajectories can be extracted.

Prior work such as Jaillet et al. (2010) addresses path planning under the consideration of a cost function defined over the configuration space. In such work, a user-given cost function is defined over the configuration space as an additional input to the standard path-planning problem. Cost functions can take into account many different aspects such as safety distances and visibility (Mainprice et al., 2011) in particular. We show how our trajectory classification method can be applied within this general costmap-based problem formulation, so as to obtain solution classes dependent both on the topology of the configuration space and the cost function defined over it. A related line of work involves characterising environmental uncertainty probabilistically and then posing sampling-based motion planning in terms of paths that exceed a specified probability threshold in terms of robustness, e.g. Aoude et al. (2013) and Luders et al. (2010). Our formulation of trajectory classification is also related to this approach since the sub or super-levelsets of a probability density function can be used with our approach.

3. Theoretical Background

Our work builds on techniques from the field of algebraic topology (Hatcher, 2002; Munkres, 1984) which studies the properties of topological spaces by constructing algebraic objects, such as the homology groups which we will discuss shortly. Special cases of topological spaces are, for example, smooth manifolds, arbitrary subsets of a vector space as well as arbitrary metric spaces. Crucially, topological properties are independent of metric information. In mathematics, algebraic topology has played a key role in the classification smooth manifolds. Homology groups in particular are homotopy invariants, meaning that homology groups do not vary under continuous deformations, called homotopies, of the topological space under consideration. A popular example of two spaces that are homotopy equivalent are a torus and a tea-cup which, when imagined as consisting of a rubber-like material can be stretched and bent (without tearing) to deform into one another. The homology groups of these two spaces are identical and measure in particular the fact that there exists a single 'hole/tunnel' in both spaces and that each space is connected.

In this work, we will not provide a full introduction to algebraic topology, since this is not within the scope of the current paper, but we will try to define and discuss some of the key concepts – focussing mainly on providing those readers with a robotics background that might be unfamiliar with the subject with an intuition for these techniques, as well as providing sufficient detail for the readers to implement the

methods proposed here. For general persistent homology, the works of Carlsson (2009) and Edelsbrunner and Harer (2008, 2010) provide an excellent introduction, while a comprehensive discussion of algebraic topology is provided in (Hatcher, 2002; Munkres, 1984).

3.1. Simplicial Complexes

A key notion in computational algebraic topology is that of a *simplicial complex* which is used to model a large class of topological spaces of interest. We recall several basic definitions (see Edelsbrunner and Harer (2010)): $k+1$ points $v_0, \ldots, v_k \subset \mathbb{R}^d$ are called *affinely independent* if the vectors $v_1 - v_0, \ldots, v_k - v_0$ are linearly independent. A k-simplex $\sigma \subset \mathbb{R}^d$ is a convex hull of $k+1$ affinely independent points $v_0, \ldots, v_k \in \mathbb{R}^d$, $\sigma = \mathrm{Conv}(v_0, \ldots, v_k)$ and we also denote σ by $[v_0, \ldots, v_k]$. A *face* τ of a simplex $\sigma = [v_0, \ldots, v_k]$ is a convex hull of a non-empty subset of $\{v_0, \ldots, v_k\}$ and we denote this relationship by $\tau \leqslant \sigma$. Note that 2-simplices hence correspond to triangles, 1-simplices to finite line segments, and 0-simplices to points in $\mathbb{R}^d$. Using unions of such simplices, we can construct interesting topological spaces by means of a simplicial complex: A *simplicial complex* is a finite collection of simplices $\mathcal{K}$ such that $\sigma \in \mathcal{K}$ and $\tau \leqslant \sigma$ implies $\tau \in \mathcal{K}$, and $\sigma, \sigma' \in \mathcal{K}$ implies $\sigma \cap \sigma'$ is either empty of a face of both σ and σ'. A *subcomplex* $\mathcal{L} \subset \mathcal{K}$ of a simplicial complex $\mathcal{K}$ is a subset $\mathcal{L}$ of the simplices of $\mathcal{K}$ such that $\mathcal{L}$ is itself a simplicial complex.

One of the aims of this work, besides our trajectory classification algorithm, is the development and use of *simplicial complexes* as a key data-structure in robotics, in particular as *a powerful mechanism to accurately represent robot configuration spaces.*

In the top left part of Fig. 2, we illustrate an example of a very simple simplicial complex in $\mathbb{R}^2$ consisting of six 2-simplices (shaded triangles), fifteen 1-simplices (edges) and nine 0-simplices (the vertices). Note that simplicial complexes are natural generalizations of graphs (which only contain 0 and 1-simplices) as well as triangulations, both of which are popular geometric data structures in robotics. The union $|\mathcal{K}|$ of all simplices in a simplicial complex $\mathcal{K}$ is a subset of $\mathbb{R}^d$ which yields a topological space whose properties can be studied using the concept of homology which we discuss now.

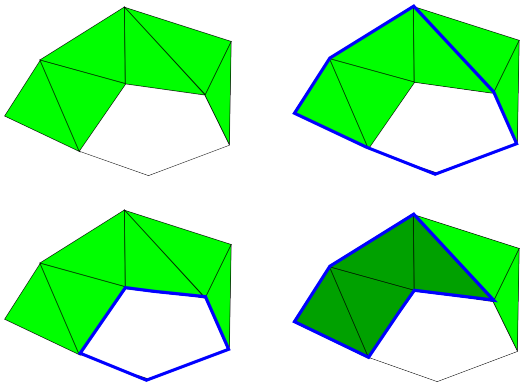

Fig. 2. We depict an example simplicial complex $\mathcal{K}$ in the top left figure. In the top right and bottom left figure, a 1-cycle c and c' is shown in blue respectively. The two 1-cycles are equivalent in the first homology group $H_1(\mathcal{K})$ since their difference is given by the shaded 2-boundary displayed in the bottom right figure.

3.2. Simplicial Homology

We consider simplicial homology over a field $\mathbb{F}$ for a simplicial complex $\mathcal{K}$. In general, non-field coefficients, such as $\mathbb{Z}$ can be considered, but for the discussion of persistent homology later, we shall require field coefficients. With field coefficients, the homology groups which we shall define shortly are simply $\mathbb{F}$-vector spaces that carry topological information. In our application, we will consider the binary field $\mathbb{F} = \mathbb{Z}_2 = \{0, 1\}$, in particular since $\mathbb{Z}_2$ coefficients can be most efficiently implemented on a computer.

A p-chain c is a formal sum $c = \sum_{i=1}^{k} \lambda_i \sigma_i$ of p-simplices $\{\sigma_1, \ldots, \sigma_k\}$ in $\mathcal{K}$ with $\lambda_i \in \mathbb{F}$ and $C_p(\mathcal{K})$ denotes the $\mathbb{F}$-vector space of all p-chains. In the case of $\mathbb{F} = \mathbb{Z}_2$, 1-chains can simply be considered as finite sets of 1-simplices (edges) and 2-chains are finite sets of 2-simplices (triangles), in particular. For every p-simplex $\sigma = [v_0, \ldots, v_p]$ let $\partial_p \sigma$ be the $p-1$-chain formed by the formal sum of all $p-1$ dimensional faces of σ as follows:

$$\partial_p \sigma = \sum_{i=0}^{p} (-1)^i [v_0, \ldots, v_{i-1}, \hat{v}_i, v_{i+1}, \ldots, v_p],$$

where $\hat{v}_i$ indicates that v_i is omitted. In the case of $\mathbb{F} = \mathbb{Z}_2$, $-1 = 1$ and the sign in the above sum can be ignored and the boundary of a 1-simplex consists of the two end-points, while the boundary of a 2-simplex consists of the three boundary edges. ∂_p extends to a linear map $\partial_p : C_p(\mathcal{K}) \to C_{p-1}(\mathcal{K})$. A chain $c \in C_p(\mathcal{K})$ such that $c = \partial_{p+1}\omega$ for some $\omega \in C_{p+1}(\mathcal{K})$ is called a *p-boundary*, and we call

c a *p-cycle* if $\partial_p c = 0$. The set of p-boundaries and p-cycles is denoted by $B_p(\mathcal{K})$ and $Z_p(\mathcal{K})$ respectively and $B_p(\mathcal{K}) \subseteq Z_p(\mathcal{K})$ since $\partial_p \partial_{p+1} = 0$. The quotient vector space $H_p(\mathcal{K}) = Z_p(\mathcal{K})/B_p(\mathcal{K})$ is called the p^{th} homology group of $\mathcal{K}$. We denote the equivalence class of a p-cycle c in the quotient vector space $H_p(\mathcal{K})$ by $[c]$.

To illustrate these concepts, consider Fig. 2: The top right figure displays a 1-cycle c over the field $\mathbb{Z}_2$ in blue. Closed loops correspond to 1-cycles because the each vertex is included an even number of times in a set of edges in the cycle. The bottom left figure corresponds to another 1-cycle c'. Observe that, when we add $c + c'$ in the $\mathbb{Z}_2$-vector space of 1-chains, we obtain the blue closed curve γ surrounding the shaded area in the bottom right figure. This curve γ is in fact a 1-boundary, since it arises as the boundary $\gamma = \partial_2 \omega$ of the shaded collection of triangles ω which form a 2-chain. Note that it is easy to see why the interior edges of these triangles are not part of the boundary: they occur precisely twice each and are hence zero modulo $\mathbb{Z}_2$. The boundary operator hence exactly assigns the geometric boundary to our 2-chain ω. The same also holds more generally for 2-chains over a general field $\mathbb{F}$, when the correct orientation is taken into account (Hatcher, 2002).

With this intuition at hand, we finally observe that $c - c' = \gamma = \partial \omega$, so that c and c' differ only by a 1-boundary. Hence, in homology $[c] = [c']$, while $[\gamma] = 0$ in homology since it arises as the boundary of the shaded triangles. We furthermore have $[c] \neq 0$ because there does not exist any 2-cycle ν such that $c = \partial_2 \nu$. To summarize: two p-cycles represent the same homology class if they differ by a p-boundary, and a homology class $[c] \in H_p(\mathcal{K})$ is non-trivial precisely if c does not arise as the boundary of some $p+1$-chain.

We denote $b_p(\mathcal{K}) = dim(H_p(\mathcal{K}))$, which is called the p^{th} Betti number of $\mathcal{K}$. In the example just considered, the fact that there exists only a single enclosed tunnel/void in $|\mathcal{K}|$ is measured by $b_1(\mathcal{K}) = 1$. Similarly, $b_0(\mathcal{K})$ is equal to the number of connected components of $\mathcal{K}$, so in our example $b_0(\mathcal{K}) = 1$ and higher Betti numbers are zero. These Betti numbers capture crucial topological information about the space $|\mathcal{K}|$ which do not depend on the particular simplicial representation of $|\mathcal{K}|$ that was chosen. Furthermore it is a remarkable fact and one of the key theorems of algebraic topology that these homology groups capture topological information which remains invariant under continuous deformations of $|\mathcal{K}|$ called homotopies (Hatcher, 2002).

3.3. Delaunay-Čech Complexes

In the preceding discussion, we assumed that a simplicial complex $\mathcal{K}$ was provided to us. In this section we discuss a particular family of simplicial complexes which can be constructed from point-cloud data in $\mathbb{R}^d$.

Consider a set of uniformly sampled points $X = \{x_1, \ldots, x_n\} \subset Y$ from a subset $Y \subseteq \mathbb{R}^d$. The r-neighborhood $X_r = \bigcup_{i=1}^n \mathbb{B}_r(x_i)$, where $\mathbb{B}_r(x_i) = \{x \in \mathbb{R}^d : \|x - x_i\| \leqslant r\}$ for $r \geqslant 0$, forms an interesting topological space. Unfortunately, X_r is not itself a simplicial complex, but we can compute the homology of X_r by representing X_r by any simplicial complex $\mathcal{K}_r$ which is homotopy equivalent to X_r. A well-known simplicial complex which has this property is the Alpha complex $A_r(X)$ of Edelsbrunner (1992) which can be defined in arbitrary dimension and is a sub-complex of the Delaunay triangulation of X. In this work, we shall instead use another recently studied simplicial complex, called the Delaunay-Čech complex (Bauer and Edelsbrunner, 2014) which is also a sub-complex of the Delaunay triangulation.

To proceed, let us first generalize the notion of a simplex and simplicial complex to that of an abstract simplex and abstract simplicial complex: an abstract k-simplex σ is just a set of $k+1$ elements, and we call k the dimension of σ. An abstract simplicial complex $\mathcal{K}$ is a finite non-empty set of abstract simplices such that if $\sigma \in \mathcal{K}$ and $\emptyset \neq \tau \subseteq \sigma \in \mathcal{K}$, then $\tau \in \mathcal{K}$. If $\tau \subseteq \sigma \in \mathcal{K}$, τ is called a face of σ (Edelsbrunner and Harer, 2008).

The Čech complex C_r with parameter $r \geqslant 0$ is an example of an abstract simplicial complex, defined by $C_r(X) = \{\sigma \subseteq X : \cap_{x \in \sigma} \mathbb{B}_r(x) \neq \emptyset\}$. Note that $C_r(X)$ has no direct representation as a subset of $\mathbb{R}^d$ and hence does not directly form a geometric simplicial complex as previously considered.

Given X, we can instead consider the complex $D(X) = \{\sigma \subseteq X : \cap_{x \in \sigma} V_x \neq \emptyset\}$ (the *Delaunay triangulation of* X) where V_x denotes the Voronoi cell containing x. We consider $D(X)$ for points in X which are in general position, which occurs with probability one and which can also

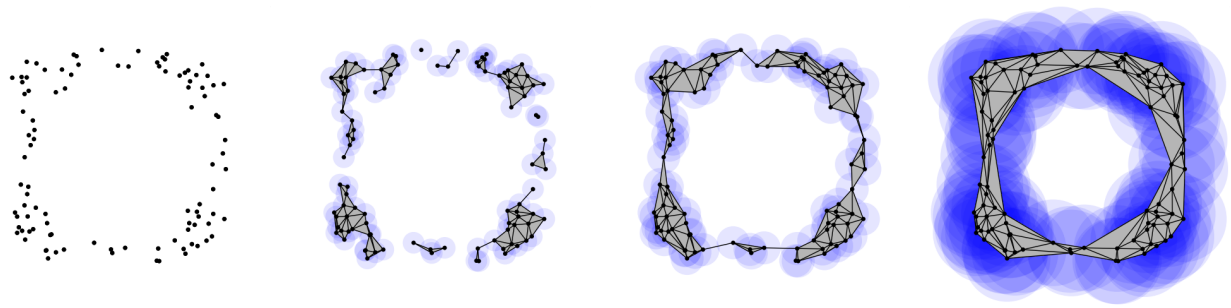

Fig. 3. We illustrate a point-cloud $X \subset [0,1]^2 \subset \mathbb{R}^2$ (left figure) as well as the resulting Delaunay-Čech complex $DC_r(X)$ at filtration parameter $r = 0, r = 0.05, r = 0.076, r = 0.2$ (from left to right). For each filtration parameter value $r \geqslant 0$, $DC_r(X)$ is topologically equivalent (homotopy equivalent) to the union of balls X_r of radius r around the samples, which is depicted in blue. Using $DC_r(X)$ we can hence compute the homology groups of X_r at all scales.

be enforced by a small perturbation of X. Note here that $D(X)$, as defined above, is a simplicial complex containing 0-simplices up to d-simplices in $\mathbb{R}^d$.

The Delaunay-Čech complex $DC_r(X)$, for $r \geqslant 0$, is the subcomplex of $D(X)$ defined by $DC_r(X) = \{\sigma \in D(X) : \cap_{x\in\sigma}\mathbb{B}_r(x) \neq \emptyset\}$. The recent work Bauer and Edelsbrunner (2014) establishes that $DC_r(X)$ is homotopy equivalent to X_r, so that topological information about X_r can be extracted from $DC_r(X)$ directly. We define $f : D(X) \to \mathbb{R}$ by $f(\sigma) = \min\{r : \cap_{x\in\sigma}\mathbb{B}_r(x) \neq \emptyset\}$, so that $DC_r(X) = f^{-1}((-\infty, r])$ and $DC_r(X)$ changes only at finitely many $r_1 < \ldots < r_m$ which can be computed at all scales by determining $f(\sigma)$ for each simplex $\sigma \in D(X)$. Every k-simplex $\sigma = \{v_0, \ldots, v_k\} \in DC_r(X)$ corresponds to the geometric simplex given by the convex hull $\mathrm{Conv}(\sigma)$, so that 0-simplices are points, 1-simplices are edges, and 2-simplices are triangles.

In Fig. 3, we illustrate a point-cloud $X \subset \mathbb{R}^2$ as well as the associated $DC_r(X)$ for several parameters $r \geqslant 0$. The previously discussed Fig. 1 also displayed a Delaunay-Čech complex constructed from sampled points. Note that, instead of $DC_r(X)$, we could also have considered the Alpha complexes $A_r(X)$ of Edelsbrunner (1995) since $A_r(X)$ is also homotopy equivalent to X_r and furthermore $A_r(X) \subseteq DC_r(X)$, and for sufficiently large parameter R the complex $A_R(X)$ and $DC_R(X)$ reach their maximal size, and $A_R(X) = DC_R(X) = D(X)$. Since $DC_r(X)$ does not change for $r \geqslant R$, we shall use the notation $DC_R(X) = DC_\infty(X)$. An advantage of $DC_r(X)$ over $A_r(X)$ in our application is that we can compute the filtration values for the 2-skeleton of $DC_r(X)$, denoted by $DC_r^2(X)$ and consisting only of simplices of $DC_r(X)$ up to dimension 2, directly without having to first compute the filtration values for the higher skeleta.

3.4. Filtrations

Consider a simplicial complex $\mathcal{K}$ and a function $f : \mathcal{K} \to \mathbb{R}$ defined on the simplices of $\mathcal{K}$. If f satisfies $f(\tau) \leqslant f(\sigma)$ whenever $\tau \leqslant \sigma$, then we observe that $\mathcal{K}_r = f^{-1}((-\infty, r])$ is a simplicial complex and $\mathcal{K}_r \subseteq \mathcal{K}_{r'}$ whenever $r \leqslant r'$, yielding a *filtration of simplicial complexes with filtration parameter* r. In the previous section, $DC_r(X)$, for $r \geqslant 0$, provides an example of such a filtration. Considering Fig. 3, we observe that as r is increased, the topology of the union of balls space X_r changes. Initially there exists one connected component per data-point. These components then merge as r increases and we can recover the fact that the point-cloud 'looks like there exists a hole in the middle' in the third and fourth figure, where the first homology group of $DC_r(X)$ is one-dimensional. Persistent homology, which we shall discuss now allows us to rigorously make statements about which homological features exist at various filtration parameters.

3.5. Persistent Homology

For a filtration of simplicial complexes, where $f : \mathcal{K} \to \mathbb{R}$, $\mathcal{K}$ is a finite simplicial complex, and $\mathcal{K}_r = f^{-1}((-\infty, r])$, we denote the finitely many filtration values at which $\mathcal{K}_r$ changes by $r_1 < \ldots < r_m$. The inclusion $\alpha_i^j : \mathcal{K}_{r_i} \to \mathcal{K}_{r_j}$, for $i \leqslant j$, induces a linear map $h_i^j : H_p(\mathcal{K}_{r_i}) \to H_p(\mathcal{K}_{r_j})$. We say that a homology class $\alpha \in H_p(\mathcal{K}_{r_i})$ is

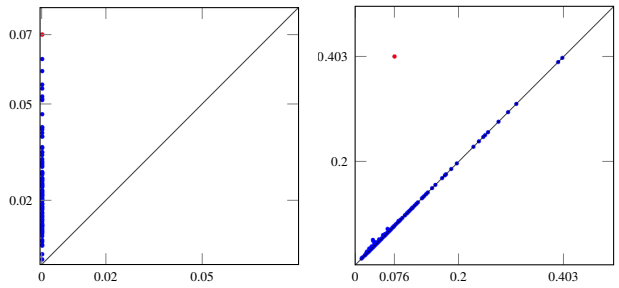

Fig. 4. The left figure displays the inessential intervals of the zeroth persistence diagram for the point-cloud displayed in the left part of Fig. 3. Each point is of the form $(0, r_i)$ and corresponds to a connected component that merges with another component at filtration value r_i. Besides the displayed points, there exists an additional essential point $(0, \infty)$ corresponding to the final connected component persisting for all filtration values. Note in particular that, for $r > 0.07$, only this essential component survives when the red persistence interval $(0, 0.07)$, marked in red, has died and $DC_r(X)$, for $r > 0.07$ has only a single connected component corresponding to $(0, \infty)$. The right figure displays the first persistence diagram. Note how the fact that a single hole exists in the data corresponds to a single persistence interval $(0.076, 0.403)$, with large persistence marked in red. Observe also how, in Fig. 3, at $r = 0.076$, the hole is first enclosed by X_r and by simplices. Similarly, at 0.403 this hole is filled for the first time.

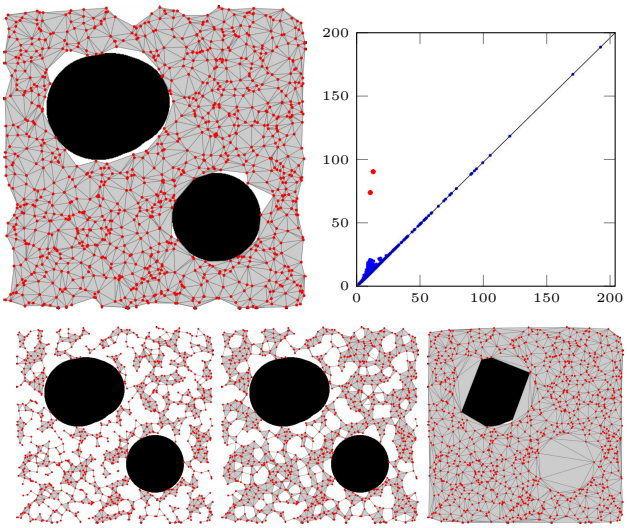

Fig. 5. A reconstruction of $\mathcal{C}_f$ with two obstacles in black and from 1000 collision free samples (red points) on a square of side-length 500. $DC_{25}(X)$ is displayed in the top-left yielding a good approximation to $\mathcal{C}_f$. The first persistence diagram for $DC_r(X)$ is shown in the top-right. The two marked red points $p_1 = (10.58, 74.0)$, $p_2 = (12.97, 90.38)$ with large persistence correspond to the birth and death filtration of the two holes in $\mathcal{C}_f$. The bottom row displays $DC_{10.58}(X)$, $DC_{12.97}(X)$ and $DC_{74.0}(X)$ which correspond to the birth of the smaller and larger hole (the first time they are enclosed by edges), and finally to the death filtration value of the smaller hole (the hole is covered at $r = 74.0$).

born at r_i if $\alpha \notin im(h_{i-1}^i)$. A class $\alpha \in H_p(\mathcal{K}_{r_i})$ born at r_i is said to die at r_j if $h_i^{j-1}(\alpha) \notin im(h_{i-1}^{j-1})$, but $h_i^j(\alpha) \in im(h_{i-1}^j)$. The difference $r_j - r_i$ is called the *persistence* of α: it measures how long a homological feature survives in the filtration. Classes born at r_i which do not die are associated to (r_i, ∞) and are called essential. The remaining classes are called inessential. Similarly, if a cycle represents an essential (inessential) class, we call the cycle essential (inessential). For $i \leqslant j$, the *p-th persistent homology group* is defined as $H_p^{i,j} = Z_p(\mathcal{K}_{r_i})/(B_p(\mathcal{K}_{r_j}) \cap Z_p(\mathcal{K}_{r_i}))$. Non-trivial elements of $H_p^{i,j}$ correspond to equivalence classes of p-cycles born at or before r_i and which *persist*, *i.e.* do not die in the filtration for $r \in [r_i, r_j)$. For $i = j$, this recovers the usual notion of homology $H_p^{i,i} = H_p(\mathcal{K}_{r_i}) = Z_p(\mathcal{K}_{r_i})/B_p(\mathcal{K}_{r_i})$. A graphical representation is obtained by the *p-th persistence diagram* which associates (r_i, r_j) to classes born at r_i and dying at r_j and (r_i, ∞) to essential classes born at r_i (with multiplicity). The number of points in $(-\infty, r_i] \times (r_j, \infty]$ equals $dim(H_p^{i,j})$ and the vertical distance of a point to the diagonal indicates how long the feature persists (see Edelsbrunner and Harer (2008)).

In Fig. 4, we display the zeroth and first persistence diagram corresponding to the point-cloud shown in Fig. 3, allowing us to understand how the number of connected components and holes in the point-cloud evolve over all filtration values. The fact that a large hole exists in the middle of this point-cloud is recovered by the existence of the single large persistence interval in the first persistence diagram. In Fig. 5, we display the first persistence diagram and $DC_r(X)$ for $DC_r(X)$ arising from collision free samples in a simple configuration space with two obstacles shown in black. Observe that the two obstacles correspond to the two red points in the diagram which are far from the diagonal. The remaining points correspond to holes which are due to noise and which do not persist for a large filtration interval.

3.6. Computation via Matrix Reduction

To compute the persistence diagrams of a filtration $\mathcal{K}_{r_1} \subset \mathcal{K}_{r_2} \subset \ldots \subset \mathcal{K}_{r_m}$, it is convenient to refine the filtration as follows: we pick an ordering $\sigma_1, \ldots, \sigma_n$ of the simplices of $\mathcal{K}_{r_m}$ such that, for all $i \in \{1, \ldots, n\}$, $K_i = \cup_{l=1}^i \sigma_l$ is a

simplicial complex and there exist indices $1 \leqslant i_1 < i_2 < \ldots < i_m = n$ such that $K_{i_j} = \mathcal{K}_{r_j}$. Such a *simplexwise filtration* can be obtained by inserting simplices in $\mathcal{K}_{r_i}$ before simplices in $\mathcal{K}_{r_j}$ if $i < j$ and by inserting the faces $\tau \subset \sigma$ of any simplex σ before inserting σ itself (Edelsbrunner and Harer, 2008).

Let $K = \bigcup_{i=1}^{n} \sigma_i$ be such a simplexwise filtration. The boundary operator $\partial : \oplus_{p=0}^{d} C_p(K) \to \oplus_{p=0}^{d} C_p(K)$ is a linear map which we express in the ordered basis $\sigma_1, \ldots, \sigma_n$ yielding an $n \times n$ matrix D with $\mathbb{Z}_2$ entries. For a matrix M, we denote by M_j the j^{th} column and by M_{ij} the (i,j)-entry. Note that D is upper triangular and $D_{ij} = 1$ if σ_i is a codimension 1 face of σ_j. We let $low(M_j) = \max\{i : M_{ij} \neq 0\}$ if $M_j \neq 0$ and $low(M_j)$ is undefined otherwise. A left-to-right column addition $M_j \leftarrow M_j + M_i$, $i < j$ is called *reducing* if it decreases $low(M_j)$ and M is called reduced if no reducing left-to-right column addition can be performed on any of its columns. The standard persistence algorithm Edelsbrunner and Harer (2010) applies left-to-right column additions to D until D is reduced, yielding a reduced matrix R. Note that this reduction approach is very similar to the standard Gaussian elimination algorithm, although performed over $\mathbb{Z}_2$ and not including all reduction steps. The algorithm can further be generalized to $\mathbb{F}$ coefficients for an arbitrary field $\mathbb{F}$.

We can keep track of the column additions by initializing the algorithm with $R = D$, $V = I_n$, so that $R = DV$. For each left-to-right column addition $R_j \leftarrow R_j + R_i$ for $i < j$, we perform the column addition $V_j \leftarrow V_j + V_i$. This algorithm terminates when R is reduced and we have $R = DV$, where V is the matrix relating R to its unreduced version D. One defines (Chen and Kerber, 2011) $P = \{(i,j) : R_j \neq 0 \text{ and } i = low(R_j)\}$, $E = \{i : R_i = 0 \text{ and } low(R_j) \neq i \text{ for all } j \in \{1, \ldots, n\}\}$. Returning to $\mathcal{K}_r = f^{-1}((-\infty, r])$, each $(i,j) \in P$ with $dim(\sigma_i) = p$ corresponds to $(f(\sigma_i), f(\sigma_j))$ in the p-th persistence diagram and is generated by the p-cycle R_j which dies with the introduction of the simplex σ_j. Similarly, each $i \in E$ with $dim(\sigma_i) = p$ corresponds to $(f(\sigma_i), \infty)$ and the p-cycle V_i which is still alive in the final filtration $K_n = \mathcal{K}_{r_m}$. Note that the cycles V_i, R_j do not correspond to canonical choices, but the persistence diagrams determine the ranks of all persistent homology groups.

3.7. $H_1(Y)$ and Homotopy Classes of Trajectories

The final piece of background work we require is the connection between the first homology group and homotopy classes of paths in a topological space Y. The obvious case to keep in mind is $Y = \mathcal{C}_f \subset \mathbb{R}^d$. Recall that the first fundamental group $\pi_1(Y, x_0)$ (Edelsbrunner and Harer, 2010) is a well-known group whose elements consist of equivalence classes of closed continuous curves through $x_0 \in Y$ and lying entirely in Y. Two closed paths $\alpha, \beta : [0,1] \to Y$ through x_0 lie in the same equivalence class if there exists a homotopy (i.e. a continuous deformation) between them which is constant at the base-point x_0. When Y is path-connected, $\pi_1(Y, x_0)$ is independent of the chosen base-point x_0 and hence often denoted simply by $\pi_1(Y)$. Furthermore, if the spaces Y, Y' are homotopy equivalent spaces, $\pi_1(Y)$ and $\pi_1(Y')$ are isomorphic as groups. Two paths γ_1, γ_2 in Y with the same start point x and end point y can be deformed into each other via a homotopy if the closed curve γ following γ_1 from x to y and then γ_2 from y to x is trivial in $\pi_1(Y)$. Hence, $\pi_1(Y)$ is a natural group to consider for the purpose of trajectory classification. Unfortunately, to the best of our knowledge, currently no sufficiently efficient method for general configuration spaces exists to compute the group structure of $\pi_1(Y)$ (*e.g.* for large scale simplicial complexes), since $\pi_1(Y)$ can be complicated and non-commutative. To extract topological information about homotopy classes, we can turn to the first singular homology group $H_1(Y)$ with binary $\mathbb{Z}_2 = \{0,1\}$ coefficients, yielding a vector space which can be explicitly computed via simplicial homology when Y is homotopy equivalent to a simplicial complex $\mathcal{K}$. The closed curve γ can be represented explicitly as a 1-cycle in a sufficiently fine subdivision of $\mathcal{K}$ when a deformation retraction from Y to $\mathcal{K}$ is computable, and γ then corresponds to a vector $[\gamma]$ in $H_1(Y) \cong H_1(\mathcal{K})$. Finally, $[\gamma] \neq 0$ implies that γ_1 and γ_2 are not homotopy equivalent, allowing us to discern homotopy classes of continuous paths. Note however that homology is a weaker concept than homotopy, so $[\gamma] = 0 \in H_1(Y)$ does not imply that γ_1 and γ_2 are homotopy equivalent. To gain somewhat more granularity, one can further replace $\mathbb{Z}_2$ coefficients for example with $\mathbb{Z}_p$ coefficients for a large prime p. In this work we choose $\mathbb{Z}_2$ coefficients due to their computational advantages for large simplicial complexes.

4. Methodology

We consider a configuration space $\mathcal{C} \subset \mathbb{R}^d$ and the set $\mathcal{C}_f \subseteq \mathcal{C}$ of collision-free configurations. We do not assume that we have an explicit description of $\mathcal{C}_f$ or $\mathcal{C}$ available, and we would like to study homotopy classes of a set of trajectories $T = \{\gamma_1, \ldots, \gamma_k\} \subset \mathcal{C}_f$ with a fixed starting point $x \in \mathcal{C}_f$ and end point $y \in \mathcal{C}_f$. In order to classify the trajectories, we shall exploit the connection between homotopy classes and the first homology group which we just discussed. We now consider two multiscale settings:

1) X is a sufficiently dense sample We assume that $X = \{x_1, \ldots, x_n\} \subset \mathcal{C}_f$ yields a sufficiently dense sample, for example sampled via rejection sampling from the uniform distribution on $\mathcal{C}$, or via a randomized exploration of the configuration space. We can then ask about a likely approximation of $\mathcal{C}_f$ from X. Our working hypothesis is that the family of spaces $\{X_r = \bigcup_{x \in X} \mathbb{B}_r(x) : r > 0\}$ contain good such estimates. If X was sampled uniformly and $\mathcal{C}_f$ is a smooth compact submanifold $M \subset \mathbb{R}^d$, this intuition is in fact well-founded due to the reconstruction theorem of Niyogi et al. (2008) which guarantees that, for a sufficiently dense sample set, X_r deformation retracts to the manifold M for appropriately chosen r. Using the previously introduced Delaunay-Čech complex and the fact that $DC_r(X)$ is homotopy equivalent to X_r (Bauer and Edelsbrunner, 2014), we will then compute homological information about X_r from $DC_r(X)$. This approach is currently applicable mostly in lower dimensions, since the 'curse of dimensionality' phenomenon implies that an increasingly large sample set would be required in higher dimensions to faithfully reconstruct $\mathcal{C}_f$.

2) $X \subset T$ Here, we assume only the availability of the trajectories T. We then discretize each trajectory γ_i as a piecewise linear curve and use the vertex positions of all the piecewise linear segments in T as our sample set X. We study the homotopy classes of these trajectories *within the topological spaces* X_r which constitute an approximation of the r-neighborhoods around T. This then allows us to classify trajectories within X_r. In this framework, holes can arise either due to obstacles in the configuration space (as in the dense case), or due to the distribution of the trajectories in $\mathcal{C}_f$. We consider applications of this case in our experiments with a Baxter robot. Observe that this approach suffers to a lesser degree from the 'curse of dimensionality', especially when approximations of X_r, such as Vietoris-Rips or Witness complexes, which we shall discuss in the following, are considered.

For a sample set X, let R be the minimal $r > 0$ such that $\gamma_i \subset X_r$ for all $i \in \{1, \ldots, k\}$. Our approach in both cases above will now be to study the homotopy classes of these paths *in the topological spaces* $X_r \simeq DC_r(X)$, *for* $r \geqslant R$.

4.1. Trajectory Discretization

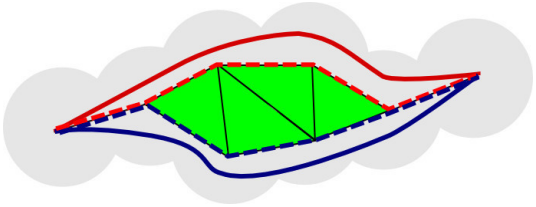

Fig. 6. We display two continuous trajectories in red and blue (solid lines) and their homotopy equivalent discretizations as sequences of 1-simplices (as dashed lines).

In order to compute properties of a trajectory $\gamma : [0, 1] \to \mathcal{C}_f$, we first need to represent γ by a homotopy equivalent path of edges (*i.e.* 1-simplices) in the simplicial complex used to model X_r. Figure 6 illustrates this in an example. Starting with a collection of samples X formed by the centers of the indicated shaded gray balls, we consider the space X_R, where R is the radius of the shaded balls. For an arbitrary continuous trajectory in X_R such as the smooth trajectories in red and blue in the figure, we require a homotopy equivalent trajectory of 1-simplices in the simplicial complex $DC_R(X)$ homotopy equivalent to X_R. In the figure these are indicated by red and blue dashed trajectories respectively.

A fast heuristic to obtain a discretization of a trajectory γ is to consider $v_i = \gamma(i/N)$, for some large $N \in \mathbb{N}$, to map v_i to a closest 0-simplex $v_i' \in DC_R(X)$ and to then replace the path segment between v_i, v_{i+1} by a shortest edge-path between v_i' and v_{i+1}' in the 1-skeleton of the simplicial complex. Alternatively, one can attempt to construct an explicit deformation retraction from X_R to $DC_R(X)$ mapping γ first to a path contained in $DC_R(X)$ and then approximating γ by a homotopy equivalent sequence of 1-simplices on a sufficiently fine subdivision of $DC_R(X)$. The Alpha complexes $A_r(X)$ of Edelsbrunner (1995) are subcomplexes of

$DC_r(X)$ for all $r > 0$ which are also homotopy equivalent to X_r and onto which an explicit such deformation retraction from X_r has been described in Edelsbrunner (1995), for example. While the study of efficient and theoretically sound homotopy equivalent trajectory discretizations could be explored further, we will instead focus on the classification problem here, assuming that each trajectory has been discretized as a path of edges in $DC_R(X)$.

4.2. Computation of $DC_r(X)$ and alternative complexes

As previously outlined, our key motivation for using Delaunay-Čech complexes in this work is the fact that $DC_r(X)$ is homotopy equivalent to the natural union of r-balls space X_r. A drawback of this approach is however the fact that we rely on the computation of a Delaunay triangulation of $X \subset \mathbb{R}^d$, which is computationally challenging as the dimension d increases. The worst case arises for n points distributed along the curve $\gamma(t) = (t, t^2, \ldots, t^d)$, called the *moment curve*, in which case the complexity of the Delaunay triangulation as measured by the number of simplices is $O(n^{\lceil d/2 \rceil})$ (Amenta et al., 2007; McMullen, 1970), while, on the other hand the complexity is $O(n)$ with a constant factor exponential in dimension for points that are uniformly distributed in a unit ball (Dwyer, 1991). Given these results and the state of the art implementations provided in (Barber et al., 1996; CGAL, 2013), Delaunay-based simplicial complexes currently face a serious 'curse of dimensionality' challenge as d is increased. In this work, we will demonstrate the feasibility of Delaunay-Čech complexes in dimensions up to 6, but for very high dimensional data, other approximations for X_r are required.

The Vietoris-Rips complex, in particular, is suitable for such data. The Vietoris-Rips complex $V_r(X)$ for a set of samples $X \subset \mathbb{R}^d$ and with parameter $r \geqslant 0$ is an abstract simplicial complex defined as follows:

$$V_r(X) = \{\sigma \subseteq X : diam(\sigma) \leqslant 2r\},$$

where $diam(\sigma)$ denotes the supremum over the pairwise Euclidean distances between points in σ. The Vietoris-Rips complex can be efficiently constructed in high dimensions *if the maximal r parameter is chosen sufficiently small* (Zomorodian, 2010). Note however that, for sufficiently large r any subset of X will be included in $V_r(X)$, resulting in a complex that is infeasible to compute even for moderately sized X. Furthermore, the Vietoris-Rips complex is not always homotopy equivalent to X_r, but provides instead an approximation (Carlsson, 2009). If computational efficiency is required at a cost of the accuracy of reconstruction, sparse approximations to $V_r(X)$ can be considered, such as the Witness complexes discussed by Carlsson (2009).

To compute 2-skeleta of $DC_r(X)$, we first compute a Delaunay triangulation using (Barber et al., 1996; CGAL, 2013). This results in d-simplices $[v_0, \ldots, v_d]$ from which we extract all 0-simplices v_i, 1-simplices $[v_i, v_j]$, $0 \leqslant i < j \leqslant d$ and 2-simplices $[v_i, v_j, v_k]$, for $0 \leqslant i < j < k \leqslant d$ without repetitions. The 0-simplices are given by vertices, and for the large simplicial complexes in our experiments, we found that extracting these k-simplices ($k = 1, 2$) was sufficiently efficiently achieved as follows: first, we extract the k-faces ($k = 1, 2$) for each d-simplex, which results in multiple occurances of simplices that are faces of more than one simplex. Next, we lexicographically sorted these resulting simplices by vertex indices via quicksort and finally removed multiple occurances of simplices. Alternatively, if sufficient memory is available, a set data-structure could be used for this purpose. In the final step, the Delaunay-Čech filtration parameter for each simplex has to be computed. Here, 0-simplices are assigned filtration value 0, 1-simplices $[v, v']$ are assigned $\frac{d(v,v')}{2}$ and for general k-simplices we can apply the MiniBall algorithm (see Edelsbrunner and Harer (2010)) to determine $f(\sigma)$, which is particularly simple in the case of 2-simplices.

4.3. Homological Trajectory Classification

Consider a set of edge-paths $\{\alpha_0, \ldots, \alpha_m\}$ in $DC_R(X)$ starting and ending at 0-simplices $s, t \in DC_R(X)$ respectively. We consider the 1-cycle $c_{\alpha_0}(\alpha_u) \stackrel{\text{def}}{=} \alpha_0 + \alpha_u \in Z_1(DC_R(X))$, where the addition is performed over $\mathbb{Z}_2$ and we have reduced all edge-paths modulo $\mathbb{Z}_2$, so that edges occurring an even number of time are removed. Now $[c_{\alpha_0}(\alpha_u)] \neq [c_{\alpha_0}(\alpha_w)] \in H_1^{i,j} = H_1^{i,j}(DC(X))$ implies $[\alpha_u + \alpha_w] \neq 0$, so that α_u, α_w are not homotopy equivalent in $DC_r(X), R \leqslant r_i \leqslant r < r_j$, where $r_1 < \ldots < r_m$ denote the critical filtration values at which $DC_r(X)$ changes. We hence have trajectory classes $\{[c_{\alpha_0}(\alpha_0)], \ldots, [c_{\alpha_0}(\alpha_m)]\} \in$

$H_1^{i,j}$ and the class membership can be computed once we have determined a basis for $H_1^{i,j}$. Note that α_0 corresponds to the zero vector $0 = [c_{\alpha_0}(\alpha_0)]$ and, for $\mathbb{Z}_2$ coefficients, there can be up to 2^k trajectory classes for fixed s, t and i, j when $dim(H_1^{i,j}) = k$. We can now compute a basis for $H_1^{i,j}$:

Lemma 4.1. *Let $K_1 \subset \ldots \subset K_n$ be a simplexwise filtration of simplicial complexes, let $R = DV$ denote the reduced boundary matrix after applying the left-to-right reduction algorithm, and let $E_p \subseteq E$, $P_p \subseteq P$ denote those elements corresponding to p-cycles only. For $1 \leqslant i \leqslant n$, a basis of $Z_p(K_i)$ is given by $S^i = \{R_t : (s,t) \in P_p, s \leqslant i\} \cup \{V_s : s \in E_p, s \leqslant i\}$, and, for $1 \leqslant i \leqslant j \leqslant n$, the image of the set*

$$\begin{aligned} T^{i,j} =& \{R_t : (s,t) \in P_p, s \leqslant i, t > j\} \\ & \cup \{V_s : s \in E_p, s \leqslant i\} \end{aligned}$$

under the quotient map

$$q : Z_p(K_i) \twoheadrightarrow H_p^{i,j} = Z_p(K_i)/(B_p(K_j) \cap Z_p(K_i))$$

forms a basis of $H_p^{i,j}$. Finally $\#E_p = dim(H_p(K_n))$.

Proof. This follows from the reduction algorithm (Edelsbrunner and Harer, 2010). Please see the appendix for further details. □

In order to classify $\{\alpha_0, \ldots, \alpha_m\}$, we first select a simplex-wise refinement $\{K_i\}_{i=1}^n$ of the filtration given by $DC_r(X)$, $r \geqslant 0$. Next, we compute the $\mathbb{Z}_2$ coordinates of $c_{\alpha_0}(\alpha_u)$ for $0 \leqslant u \leqslant m$ in the basis S^n once. To classify trajectories *at a scale given by the filtration value* $r_i = f(\sigma)$, we simply look up the binary coordinates of $c_{\alpha_0}(\alpha_u)$ restricted to the basis elements $T^{i,i} \subseteq S^n$. Similarly, we can check if two trajectories α_u, α_w are homotopy inequivalent for all $r_i \leqslant r < r_j$ by looking up whether the coordinates of $c_{\alpha_0}(\alpha_u)$ and $c_{\alpha_0}(\alpha_w)$ differ in the basis $T^{i,j} \subseteq S^n$.

Note now that $DC_r(X) = D(X)$ for sufficiently large r, where $D(X)$ denotes the full Delaunay triangulation, and $H_1(D(X)) = \{0\}$ since $D(X)$ is contractible. Hence E_1 is empty implying that we *do not need to keep track of the matrix V* to determine a basis of $H_1^{i,j}$. This is important since, in our experiments, these matrices have up to millions of columns and R is typically very sparse and of low rank, while V has full rank. Since low is injective on the set S^n, we order elements of S^n (for $p = 1$) by their low value and we store $low^{-1} = l$ as a map such that $l(k)$ is the element $s \in S^n$ with $low(s) = k$. For any cycle $c \in Z_1(K_i)$, we can then trivially solve for the coefficients in the basis S^n by iterating $c \leftarrow c + l(low(c))$. Each iteration reduces $low(c)$ until we arrive at the zero vector. In the ordered basis S^n, c then has non-zero coefficients $F(c) \in \mathbb{Z}_2^{\#S^n}$ exactly at those basis elements $s \in S^n$ for which $low(s) = low(c)$ during the execution of the above loop. Again, n can be very large (millions), but the vector $F(c)$ is in our experiments very sparse so that the algorithm does not exhibit its worst cast $O(n^2)$ computation time. We call $F(c_{\alpha_0}(\alpha_u)) \in \mathbb{Z}_2^{\#S^n}$ the *persistent cycle coordinates* of α_u with respect to α_0.

If we want to determine a trajectory class at scales corresponding to filtration values $r_i < r_j$, we select the coordinates $F^{i,j}(c_{\alpha_0}(\alpha_u))$ of $F(c_{\alpha_0}(\alpha_u))$ corresponding to the basis $T^{i,j}$. Two trajectories α_u, α_w are then not homotopy equivalent if $F^{i,j}(c_{\alpha_0}(\alpha_u)) \neq F^{i,j}(c_{\alpha_0}(\alpha_w)$. Each non-zero coordinate of $F(c_{\alpha_0}(\alpha_u))$ corresponds to a column R_t of R which has a death filtration value $f(\sigma_t)$. At filtration value r, only those non-zero coordinates that have been born and have not died yet contribute to the classification of cycles. We hence obtain an agglomerative clustering of trajectories lying in a common $DC_R(X)$ as we increase the filtration value $r \geqslant R$. Finally, at r_m, $DC_{r_m}(X) = DC_\infty(X) = \mathrm{Conv}(X)$ and all trajectories then lie in the same class.

Illustration

Consider Fig. 1. The red trajectory corresponds to α_0, the two blue trajectories in the left and middle figure represent α_1, α_2 respectively, and all trajectories lie in $DC_{r_i}(X), r_i = 11.07$. We have $[c_{\alpha_0}(\alpha_0)] = [c_{\alpha_0}(\alpha_1)] = 0 \in H_1^{a,b}$ for all $i \leqslant a \leqslant b$, but $[c_{\alpha_0}(\alpha_2)] \neq 0 \in H_1^{a,b}$, for $i \leqslant a \leqslant b \leqslant j$, where $r_j = 73.76$ is the critical filtration value at which the hole surrounded by α_0, α_2 gets filled in.

4.4. Arbitrary filtration functions

Suppose now that we have sampled $\mathcal{C}_f$ sufficiently densely and that $DC_R(X)$, for some fixed R, provides a good approximation of $\mathcal{C}_f$. Consider a function $c : \mathcal{C}_f \to \mathbb{R}$. Our aim now is not only to classify trajectories in the

space $\mathcal{C}_f$, but to take into account a threshold for the function c. Naturally, such $c : \mathcal{C}_f \to \mathbb{R}$ can arise as a cost or risk function associating a certain cost or risk to each $x \in X$. Similarly, one might have defined a probabilistic model associating a certain likelihood or density value to each $x \in X$. Given c, we define the value $\hat{c}$ of a k-simplex $\sigma = \{v_0, \dots, v_k\} \in DC_R(X)$ to be $\hat{c}(\sigma) = \max(c(v_0), \dots, c(v_k))$. Then $\hat{c}$ satisfies $\hat{c}(\tau) \leqslant \hat{c}(\sigma)$ whenever $\tau \subseteq \sigma$. In order to apply our algorithm without having to keep track of the potentially non-sparse matrix V, we furthermore let $\hat{c}(\sigma) = \max(c(v_0), \dots, c(v_k)) + C$ for any k-simplex $\sigma \in D(X) - DC_R(X)$ and for C larger than the cost of any $v \in X$. Then $L_{R,\lambda} = \hat{c}^{-1}((-\infty, \lambda])$ yields a filtration as λ varies and $L_{R,\lambda} \subseteq DC_R(X)$ for $\lambda < C$ and $L_{R,\infty} = D(X)$, ensuring $H_1(L_{R,\infty}) = 0$. We think of $L_{R,\lambda}$ as an approximation to $M_{R,\lambda} = X_R \cap c^{-1}((-\infty, \lambda])$ for $\lambda < C$. Note that this approach yields a filtration by sub-levelsets. In the case of a probability density c, super-levelsets $c^{-1}([\lambda, \infty))$ might be more appropriate. In that case, we simply replace c by $-c$ in the above framework to yield the desired filtration.

4.5. General start and goal regions

Real-world trajectories are often noisy and varied and might not naturally start and end in a fixed start point s and end-point t. In this section, we discuss an extension of our approach which enables us to work with connected regions S, T generalizing the points s, t respectively. We hence assume that there exist connected regions $S, T \subset \mathcal{C}_f$, $S \cap T = \emptyset$ such that, each of our trajectories starts in a point in S and terminates in a point in T.

Note that, in the case where S, T correspond to subcomplexes with trivial first homology, the following procedure could be employed: we select a representative 0-simplex $s \in S$ and $t \in T$ and extend each trajectory $\{\alpha_0, \dots, \alpha_m\}$ to a trajectory from s to t by first determining an edge-path from s to $\alpha_j(0)$ inside S, then following α_j and finally concatenating the resulting path by another edge-path from $\alpha_j(1)$ to t. The resulting augmented trajectories could then be classified with our previous approach. In the case where $H_1(S) \neq 0$ or $H_1(T) \neq 0$, this procedure will however depend on how the trajectories are augmented, requiring motion planning under homology constraints. We now propose an alternative approach that can be applied in all cases and which relies only on a *cone construction*.

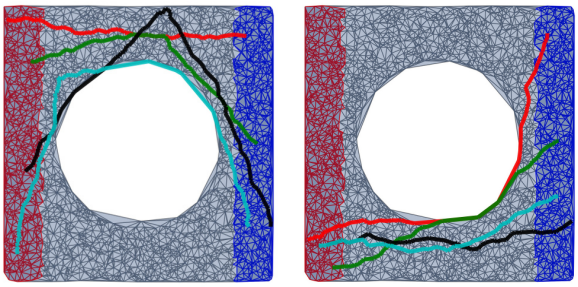

Fig. 7. Examples of trajectory classes obtained using the cone construction for samples in $[0, 1]^2 \subset \mathbb{R}^2$. All the trajectories depicted here start in the red part of the complex and terminate in the blue part of the complex. The trajectories in the left part of the figure belong to a single topological class and the trajectories to the right similarly belong to a single topological class at the depicted filtration value of $R = 0.1$. The subcomplexes corresponding to S and T are shaded in red and blue respectively. The square has a side-length one and we used 3000 sample points X. The depicted simplicial complex shows only the simplices of $\widehat{DC_R}(X)$ which do not involve the special points s, t.

Intuitively, we would like to work with a quotient space where the regions S, T are identified with points s, t respectively. We hence define a simplicial complex filtration $\widehat{DC}$ with two additional 0-simplices at filtration value zero, called s, t. We then reassign filtration value zero to any simplex in the filtration whose vertices all lie either in S or in T. We furthermore define simplices $\bigcup_{p=1}^{d+1}(S^p \cup T^p)$, where

$$S^p = \{\{v_0, \dots, v_{p-1}, s\} : \sigma = [v_0, \dots, v_{p-1}] \in DC_\infty(X), \\ v_j \in S \text{ for all } j \in \{0, \dots, p-1\}\}$$
$$T^p = \{\{v_0, \dots, v_{p-1}, t\} : \sigma = [v_0, \dots, v_{p-1}] \in DC_\infty(X), \\ v_j \in T \text{ for all } j \in \{0, \dots, p-1\}\}.$$

and also assign filtration value zero to these and we create a simplex-wise filtration where we also insert faces of simplices before their containing simplices whenever these have identical filtration value. This yields a new filtration where the two subcomplexes of $DC_\infty(X)$ corresponding to S, T and the newly added cones over these subcomplexes appear at filtration value zero and additional simplices from $DC_R(X)$ are only added in afterwards. Any simplex in $\widehat{DC_R}(X)$ which involves s, t does not correspond to a specific geometric simplex in $\mathcal{C}_f$, but is abstract in nature. Note that s corresponds to the apex of the cone over S and t to the

apex of the cone over t. Since cones are contractible topological spaces, this yields an explicit identification of the sub-complexes corresponding to S and T with the points s, t respectively.

Each trajectory α can now first be discretized as a sequence of edges in $DC_R(X)$ as before. Assume that α starts in point $a \in S \cap X$ and terminates in $b \in T \cap X$. By appending the edges (s, a) and (b, t) to each such trajectory, we obtain a set of lifted trajectories $\hat{\alpha}_0, \ldots \hat{\alpha}_m$ in $\widehat{DC}_R(X)$ which now all start and end in the same (non-geometric) points s and t respectively.

Since we will only be interested in the first homology group, we can in fact ignore all simplices of dimension larger than 2 in the above construction since these do not influence the first homology group. We can then apply our previous algorithm on these augmented trajectories to obtain a topological classification. If two trajectories $\hat{\alpha}_i, \hat{\alpha}_j$ are inequivalent at filtration $R > 0$ using this approach, then the corresponding trajectories α_i, α_j cannot be continuously deformed into one another, even when the end-points are allowed to move within the sub-complexes corresponding to S, T. Fig. 7 illustrates two examples of classes of trajectories which are considered inequivalent using this approach. The computational cost incurred by adding the cone construction is dependent only the number of additional simplices in the 2-skeleton of $\widehat{DC}$. In particular, we add two 0-simplices s, t, and one edge for each point in $S \cup T$ as well as one triangle for each edge lying either entirely in S or entirely in T.

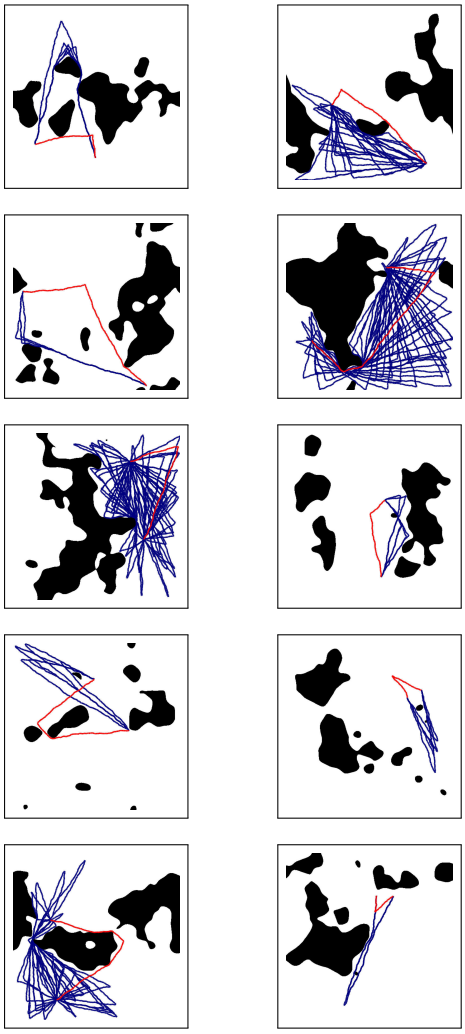

Fig. 8. We display example worlds and examples of paths which were determined to lie in a single class (in blue) at filtration value r_2. $DC_{r_2}(X)$ was constructed from 100000 samples and the classes are computed using the indicated red trajectories corresponding to α_0.

5. Experiments

Our experiments were performed on an Intel Core i7 laptop with 8GB of RAM. We present only the computation times of core algorithms and disregard the time required to load data into memory. We used the matrix and binary tree column vector data structure of the PHAT library (Bauer et al., 2013) to efficiently manipulate large boundary matrices. Instead of working with the full simplicial complex $DC_r(X)$, we extracted the 2-skeleton $DC_r^2(X)$ from the Delaunay triangulation $D(X)$. The 2-skeleton is sufficient for our purposes since $H_1^{i,j}(DC_r(X))$ does not depend on higher dimensional simplices. We reduced only the submatrix of the boundary matrix corresponding to the first homology group. $D(X)$ was computed with CGAL (2013) for all but our Baxter experiments where we used QHull Barber et al. (1996) which was faster in higher dimensions.

Trajectory classification in 2D We generated the set of 2D worlds $W_1, \ldots, W_{10}$ displayed in Fig 8 and of size 512×512 by sampling Gaussian Random Fields and defining those regions above a threshold to be obstacles. From the resulting free space $\mathcal{C}_f$, we sampled $N \in \{1000, 10000, 100000, 1000000\}$ uniform samples. We computed the Delaunay-Čech filtration for all examples and recorded the computation times of the Delaunay triangulation, for the construction of the filtration, as well as the time required to reduce the boundary matrix D to its reduced form

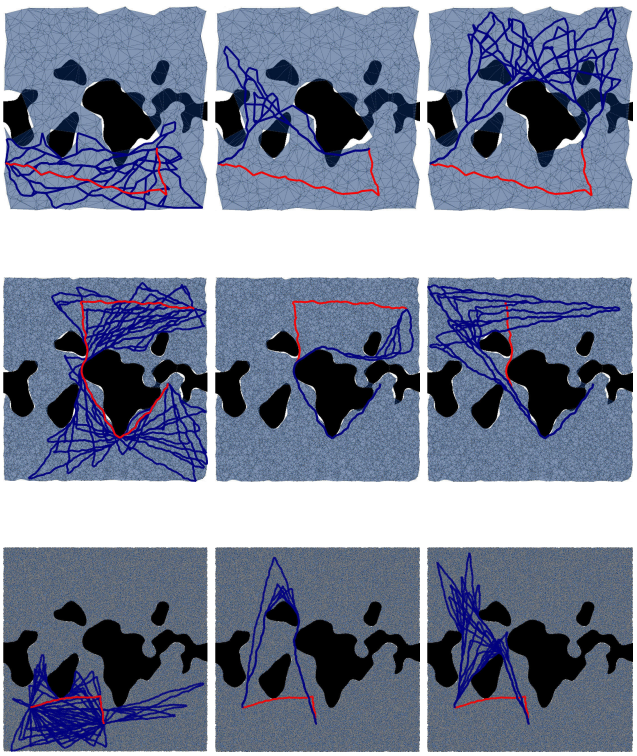

Fig. 9. We display the example world W_1 with $DC_{r_2}(X)$ for 1000, 10000 and 100000 sample points per row. In each column, we plot paths $\alpha_1, \ldots, \alpha_s$ (in blue) which belong to a fixed trajectory class at filtration value r_2. The fixed reference path α_0 is plotted in red. As expected, we can clearly see that two paths in different classes also lie in different homotopy classes. In our experiments, paths within a class are furthermore homotopy equivalent in $DC_{r_2}(X)$, but the quality of the approximation $DC_{r_2}(X) \simeq \mathcal{C}_f$ is only sufficient for 10000 or more sample points as can be seen in the right figure in the first row. There, some 2-simplices (triangles) cover the thin obstacle region to the right.

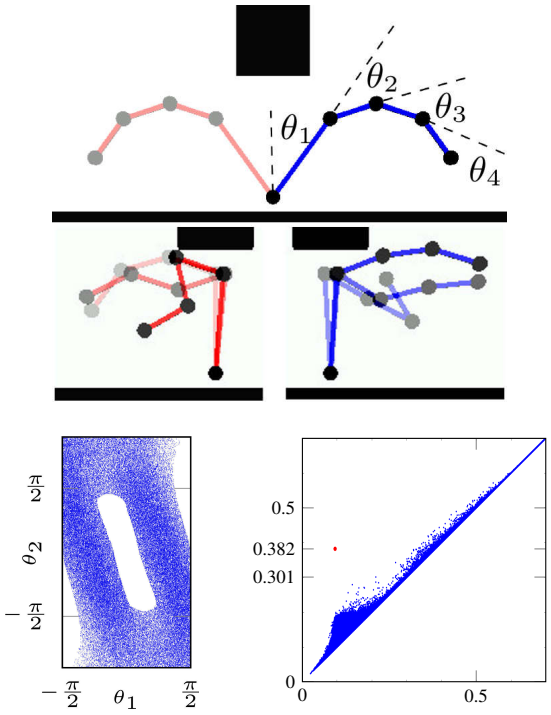

Fig. 10. The top figure displays the robot arm in start configuration (blue) on the right and in goal configuration (red) on the left. The bottom right figure displays the first persistence diagram for our reconstruction with one red point far above the diagonal. A projection of the samples onto θ_1, θ_2 is shown in the bottom left and an illustration of the difference between the two trajectory classes for $r \in [0.301, 0.382]$ is shown in the middle figure. In the first trajectory class (in red), the arm is extended to the left when passing under the narrow passage while in the second class (in blue), the arm is extended to the right.

R. The Delaunay triangulation took 1ms, 2ms, 76ms, 810ms, the construction of the filtration took 11ms, 31ms, 278ms and 3.27s and the reduction of the boundary matrix took 14ms, 13ms, 76ms, 981ms on average as the sample size increased. We investigated the filtration $DC_r(X)$ at various thresholds. At a filtration value of $r_1 = 25\sqrt{1000/N}$, we found that $\mathcal{C}_f$ was conservatively covered, while at $r_2 = 35\sqrt{1000/N}$, the space was well covered with a minimum number of holes in collision free areas. In order to investigate interesting path classes, we generated a set of 1000 paths per world and sample setting as follows: In 10 trials, we selected two sample points v_1, v_2 at random and, for each such setting, we selected another 100 random waypoints $w_1, \ldots, w_{100}$ from the sampled point-cloud. We determined shortest edge-path from v_1 to w_i and then to v_2 utilizing Dijkstra's algorithm on the 1-skeleton graph of $DC_{r_1}(X)$. The computation times for the *persistent cycle coordinates* for these paths were 1.8ms, 10ms, 115ms and 1.75s for a batch of 100 query paths and for the respective sample sizes on average. These encouraging timings suggest that our framework could be used as a classification 'black box' *e.g.* for continuous trajectory optimization engines.

Trajectory classification in 4D We consider the planar robot arm displayed in the top part of Fig. 10 which is attached to the central black disk and with 4 joints $\theta_1, \ldots, \theta_4$. We constrain $\theta_1 \in [-\frac{\pi}{2}, \frac{\pi}{2}]$, $\theta_2, \theta_3, \theta_4 \in [-0.9\pi, 0.9\pi]$ and furthermore disallow self-collisions and collisions with the environment (the black rectangle and the floor), yielding $\mathcal{C}_f \subset \mathbb{R}^4$. The robot now has the task of moving from the start configuration displayed in blue to the red goal joint

configuration as shown in the top left figure. We sampled 100000 poses uniformly in $\mathcal{C}_f$ using OpenRave (Diankov and Kuffner, 2008) and applied our framework. $DC^2_\infty(X)$ had about 6.2 million triangles and 1.8 million edges. The bottom right part of Fig. 10 displays the resulting first persistence diagram which clearly shows that a single homological feature has large persistence in $\mathcal{C}_f$. The projection of the joint configurations onto the first two angles, as shown in the middle figure, confirms the existence of a single hole. We computed 1000 edge-paths in $DC_{0.25}$ between the start and end-configuration using 1000 random waypoints as before. For filtration values $r \in [0.301, 0.382]$ only two trajectory classes existed. The reduction of the boundary matrix took 0.46s, while the persistent cycle coordinates for all 1000 paths were calculated in 0.55s. The Delaunay triangulation in $\mathbb{R}^4$ took 251s, partially due to the increased dimension. Note however that these results are not directly comparable to the 2D case since methods for 2D Delaunay triangulations in CGAL (2013) are especially optimized. We inspected the trajectories in each homology class and found that they were classified according to whether the second link was positioned to the left or to the right of the base link of the arm when $\theta_1 = 0$ as the arm passed the narrow passage (see Fig. 10). Our framework hence allows the robot to *discover* the fact that two fundamentally different solution trajectory classes exist. Note that it is also intuitively clear that no trajectory in the red class can be continuously deformed to any trajectory in the blue class since the second link would have to collide with the marked black obstacle above the arm at some point in the course of any such deformation. Our method hence allows us to automatically recover this information.

Filtrations with cost functions We consider the free configuration space $\mathcal{C}_f \subset \mathbb{R}^2$ of size 250 by 500 with two obstacles (in white) displayed in Fig. 11. We would now like to distinguish not only between homotopy classes depending on the obstacles in the configuration space, but also discern how trajectories behave with respect to the two peaks of the cost function. The simplicial complex $L_{10,\lambda}(X)$ is displayed for 10000 samples X and height values are determined by the cost function. At cost threshold $\lambda = 90$, the top of one of the hills defined by the cost-function is removed from the complex in the rightmost figure (indicated in blue), while at $\lambda = 70$ both hills are truncated in the remaining figures. We sampled 100 random paths in this configuration space by fixing the initial and terminal vertex at the start and endpoint of the drawn red reference trajectory and by sampling random waypoints as before. The figure displays example trajectory classes for differing cost filtration values. Note how, at a cost threshold of $\lambda = 90$ in the right plot, the two classes depicted in the two leftmost parts of the figure merge.

Baxter robot, 3D and 6D We now investigate a kinesthetic demonstration scenario where the Baxter robot in Fig. 12 is taught a set of trajectories which we then classify topologically. In the 1^{st} experiment (E1, Fig. 12, column 1-2), the robot is shown two ways to reach from one point above its head to a point in front of its torso. Only one arm is moved in each demonstration while the other arm remains still. We used the end-effector positions of the moving arm to represent trajectories in $\mathbb{R}^3$. In the 2^{nd} (column 3-4) and 3^{rd} (column 5-6) experiment E2 and E3, we record the positions of both end-effectors during dual arm manipulations resulting in a 6D configuration space. In E2, the robot is taught to pick up a cylindrical object with both hands from a table and to move it to one of two positions, one higher and one lower than the table. We also vary the distance between the hands during grasping between demonstrations. The trajectories in E2 are periodic. The motions start with the arms in a rest position on the sides, the object is then grasped and moved, and the rest position is visited again. In experiment E3, the robot moves the same object from a horizontal to a vertical configuration, but a metal bar is located between the robot and the object. Two intuitive motion classes are based on whether the left arm crosses in front of the obstacle, or behind it. Note that, in experiment E1 and E2, no obvious obstacles lie directly in $\mathcal{C}_f$, but due to the type of demonstrations and the robot's joint limits, the space $X_r \simeq DC_r(X) \subset \mathcal{C}_f$ exhibits interesting voids which we can exploit for classification. We found that trajectories were well-approximated using the described heuristic mapping to nearby edge-paths in $DC_R(X)$ (Sec. 4) for $R = 0.08m$ (experiment E1) and $R = 0.15m$ (experiment E2 and E3) respectively. For smaller r, $DC_r(X)$ was either not path-connected, or the edge path approximation deteriorated significantly. We hence investigated classifying paths in X_r for $r \geqslant R$. Fig. 13 displays how the number of

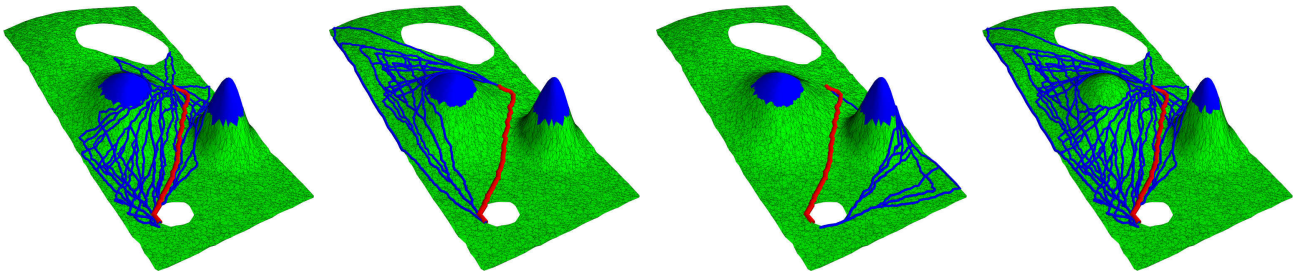

Fig. 11. We display a cost function and classes of trajectories (in blue) depending on a cost threshold and a path α_0 (in red). For the higher threshold in the rightmost plot, the two classes in the two leftmost figures merge.

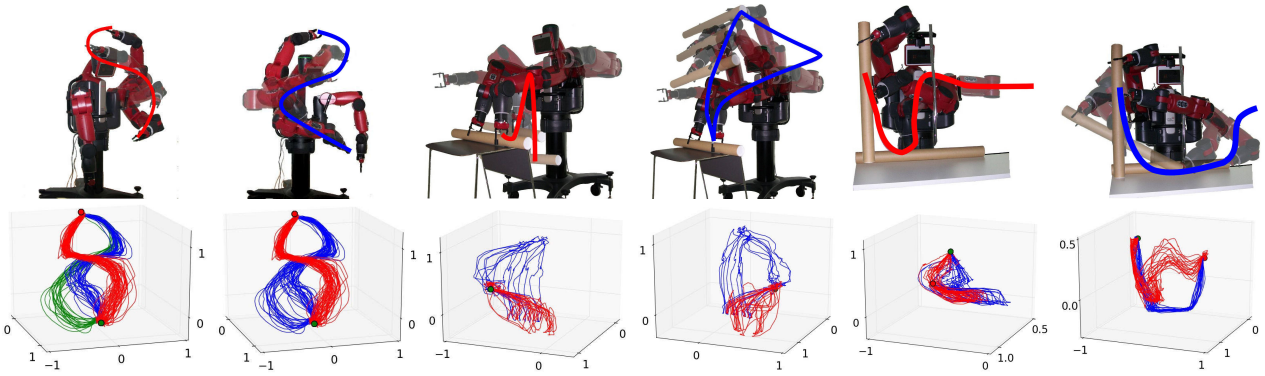

Fig. 12. Baxter robot experiments (E1: column 1-2, E2: column 3-4, E3: column 5-6). Trajectory classes are illustrated in the top row and details are provided in the bottom row. We recorded 97, 18 and 40 trajectories resulting in point clouds with 9594, 3650, 4326 points in dimension 3, 6, and 6, and in filtrations with 0.19, 2.72, 3.05 million edges and triangles for E1, E2, E3 respectively. The computation times for (Delaunay triangulation, 2-skeleton and filtration computation, boundary matrix reduction) were $(0.22, 0.26, 0.12)$, $(41.84, 46.95, 44.17)$ and $(44.92, 51.99, 52.44)$ seconds for E1, E2, E3 respectively, while the classification of all trajectories in each experiment took no more than 0.05s. Bottom row: The first two images show the end-effector trajectories in E1. While we obtain 3 classes for $r = 0.08m$ (first image) the green class merges with the blue one at $r \approx 0.084m$ (second image, see Fig 13). The 3^{rd}, 4^{th} and 5^{th}, 6^{th} image show the right and left hand trajectories for E2 and E3 respectively. The trajectory classes at $r = 0.19m$ are indicated in red and blue.

topological trajectory classes changes with varying r, and the second row of Fig. 12 displays trajectory classes at various filtration values. In all experiments, there exists a large filtration interval with just two trajectory classes corresponding to the two intuitive classes we just described for each experiment. The first image in the second row of Fig. 12 also illustrates the three classes one obtains in E1 for a choice of $r = 0.08m$. More details on the computation times and the sizes of the simplicial complexes is provided in Fig. 12.

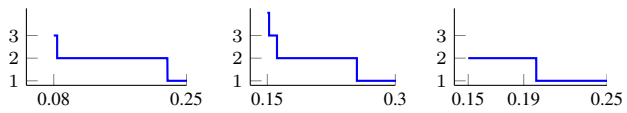

Fig. 13. Number of path classes (vertical axis) vs. filtration value (horizontal axis) for experiments E1, E2, E3 from left to right. All paths exist starting at filtration value $R = 0.08m$ for E1 and $R = 0.15m$ for E2 and E3.

Cone construction and human-robot collaboration In this experiment, we demonstrate the cone construction with real world data and provide initial experiments exploring how our approach could be incorporated in a human-robot collaboration scenario. In particular, we illustrate how, by topologically clustering trajectories, a robot can automatically extract high-level motion behaviors.

We consider the top part of Fig. 14. A PR2 robot is positioned at the depicted position in the photo and does not move, while a human stands at the green disk marked on the floor. Using a Nest of Birds magnetic motion tracker, we recorded the position of an object that is initially placed in the PR2's right gripper and which is then handed over to a human worker who then places it on the blue table by turning his upper body while keeping his feet on the shaded green disk on the floor. In the first experiment, we recorded

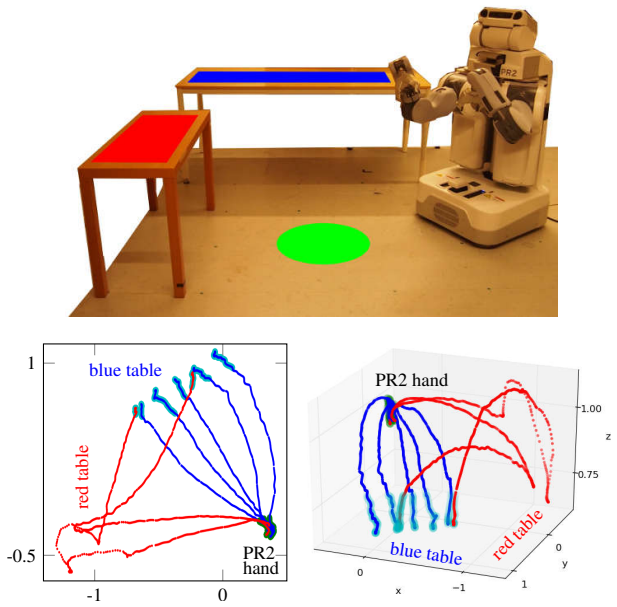

Fig. 14. The top figure displays a PR2 robot in an environment with two tables. The marked red and blue regions simulate two target work surfaces. A human is positioned to stand on the green marked disk on the floor. We recorded human-robot hand-over trajectories simulating a human worker collaborating with the robot. The 3D position of an object placed in the PR2's right endeffector is tracked and the human worker takes the object and places it on the blue work surface, but in some cases first moves the object to the red work surface for an intermediate simulated 'assembly step'. The bottom part of the figure illustrates an example topological classification of such trajectories at filtration $R \approx 16cm$. The found red trajectories correspond to sequences where the object is first placed on the red work surface, while blue trajectories correspond to sequences without this intermediate step. The bottom left figure is displayed from a bird's eye viewpoint, while the bottom right figure displays a viewpoints from behind the blue work surface.

7 trajectories of the object being handed over and placed on the blue table, resulting in 2486 points in $\mathbb{R}^3$. The bottom part of Fig. 14 indicates the trajectories. Green shaded points correspond to points which we classified as lying in the start region S in the vicinity of the robot's right gripper. The shaded cyan points correspond to goal positions T in the vicinity of the blue table-region.

We applied our cone construction and classified the trajectories. For a large filtration interval between $R \approx 16cm$ and $R \approx 21cm$, we obtained two classes as indicated in color the figure. Here, the red trajectories imitate a worker first placing the object on the table marked in red in the top

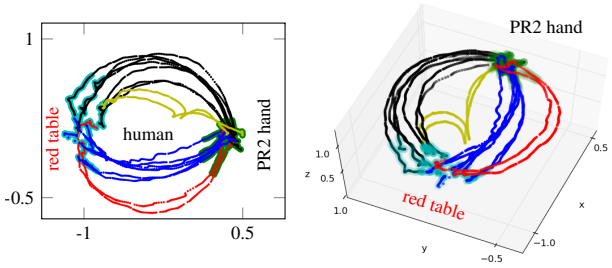

Fig. 15. Top and 3D-view of our second set of recorded trajectories between the robot's endeffector and the work surface marked in red in Fig. 14. The depicted classification was obtained at filtration $R \approx 15cm$ and uses the cone construction.

part of Fig. 14 (*i.e.* to perform an additional work step), followed by placing the object on the table marked in blue. Our method is able to identify these outliers as belonging to a seperate class (red) from the motions that move directly towards the blue table region (blue trajectories). Note also how the cone construction allows us to identify the wide variations of terminal positions of the blue trajectories.

In a second experiment, we recorded 18 trajectories resulting in 4228 points in $\mathbb{R}^3$ (and 45895 edges and 83334 triangles in $\widehat{DC}$). This time, the object is picked up from the PR2's hand and placed on the red table. The human turns either left or right by 180 degrees during this motion and does not leave the marked green spot in Fig. 14. The resulting trajectories are depicted from two perspectives in Fig. 15. Again, we apply the cone construction with points marked in green and cyan corresponding to S and T respectively. At filtration value $R \approx 15cm$, we obtain the 4 trajectory classes indicated in color. At $R \approx 20cm$ the red and blue classes merge and at $R \approx 23cm$ the green and black classes merge and only two classes exist. One corresponding to the black and green trajectories and one corresponding to the blue and red trajectories. These finally merge into one class at $R \approx 35cm$. The two classes found in the significant filtration interval $(23cm, 35cm)$ exactly correspond to whether the worker turned to the left or to the right after obtaining the object. The green trajectories correspond to a simulated case where the object was dropped or placed on the ground before being put on the table marked in red. Again, our method is able to identify such interesting trajectories as well as the high-level information about the turning behavior in this trajectory dataset. We believe that such classification information could serve as input to further algorithms to allow the

robot to reason about the behaviors of the observed human worker, for example in order to optimally assist the human worker or to automatically extract clusters of trajectories based on which dynamic motion primitives can be learned.

6. Conclusion and Future Work

In this work, we have proposed a novel *sampling-based* approach to studying topological classes of trajectories in general configuration spaces which utilizes persistent homology and filtrations of simplicial complexes as key ingredients. We have shown the applicability of our approach in real world scenarios and believe that our approach could furthermore be incorporated with many existing algorithms. For example, the integration of *local* trajectory optimization based algorithms with our approach which extracts *global* information about trajectories could be of interest. Another promising future application of our method could be a class-dependent generation of dynamic motion primitives.

While our method scaled well to large sample sizes in 2 to 4 dimensions and was applicable also for a smaller set of samples in 6D, Delaunay triangulations and hence $DC_r(X)$ have a worst-case complexity of $O(n^{\lceil d/2 \rceil})$ (McMullen, 1970) in dimension d and sample size n making triangulations in higher dimensions computationally very challenging. It remains to be seen where the boundaries of feasibility of simplicial complex based methods lie exactly in this respect in robotics. The generalization of computer graphics based works on mesh-compression such as Blandford et al. (2005) to higher dimensions might enable us to scale the presented techniques to very large complexes by reducing the required memory footprint, for example. Another direction of future research will be an investigation of alternative simplicial complexes, such as Vietoris-Rips and (weak) witness complexes (Carlsson, 2009) which approximate X_r and which could be used to mitigate the 'curse of dimensionality' in higher dimensions to some extent. Another potential direction could be the use of projections or non-linear dimensionality reduction techniques to accommodate higher dimensional data.

Our work has established that filtrations of simplicial complexes can provide a rich alternative data-structure for robotics which is capable to generalize the notion of a graph to extract novel topological information for the purpose of robotics applications. In future work, we plan to further investigate the potential applications of these data-structures and techniques in robotics.

Acknowledgments

This work was supported by the EU projects TOMSY (IST-FP7-270436) and TOPOSYS (ICT-FP7-318493).

References

N. Amenta, D. Attali, and O. Devillers. Complexity of delaunay triangulation for points on lower-dimensional polyhedra. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1106–1113. Society for Industrial and Applied Mathematics, 2007.

G. S. Aoude, B. D. Luders, J. M. Joseph, N. Roy, and J. P. How. Probabilistically safe motion planning to avoid dynamic obstacles with uncertain motion patterns. *Autonomous Robots*, 35: 51–76, 2013.

C. B. Barber, D. P. Dobkin, and H. Huhdanpaa. The quickhull algorithm for convex hulls. *ACM Trans. Math. Software*, 22(4), 1996.

U. Bauer and H. Edelsbrunner. The morse theory of Čech and Delaunay filtrations. In *Proc. of the Thirtieth Annual Symp. on Comp. Geometry*, SOCG'14, pages 484:484–484:490, New York, NY, USA, 2014. ACM.

U. Bauer, M. Kerber, and J. Reininghaus. PHAT (Persistent Homology Algorithm Toolbox), 2013. http://code.google.com/p/phat/.

S. Bhattacharya, V. Kumar, and M. Likhachev. Search-based path planning with homotopy class constraints. In *Proc. of The Twenty-Fourth AAAI Conf. on Artificial Intelligence*, Atlanta, Georgia, 11-15 July 2010.

S. Bhattacharya, M. Likhachev, and V. Kumar. Identification and representation of homotopy classes of trajectories for search-based path planning in 3D. In *Proc. of Robotics: Science and Systems*, 27-30 June 2011.

S. Bhattacharya, D. Lipsky, R. Ghrist, and V. Kumar. Invariants for homology classes with application to optimal search and planning problem in robotics. *Electronic pre-print*, Aug 2012. arXiv:1208.0573.

A. Billard, S. Calinon, R. Dillmann, and S. Schaal. Robot programming by demonstration. In *Springer handbook of robotics*, pages 1371–1394. Springer, 2008.

D. K. Blandford, G. E. Blelloch, D. E. Cardoze, and C. Kadow. Compact representations of simplicial meshes in two and three dimensions. *International journal of computational geometry & applications*, 15(01):3–24, 2005.

O. Brock and O. Khatib. Real-time re-planning in high-dimensional configuration spaces using sets of homotopic paths. In *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA'00), 2000.*, volume 1, pages 550–555. IEEE, 2000.

J. Canny. The complexity of robot motion planning. MIT press, 1988.

G. Carlsson. Topology and data. *Bull. Amer. Math. Soc. (N.S.)*, 46 (2):255–308, 2009.

CGAL. Cgal, Computational Geometry Algorithms Library, 2013. http://www.cgal.org.

F. Chazal, B. T. Fasy, F. Lecci, A. Rinaldo, A. Singh, and L. Wasserman. On the bootstrap for persistence diagrams and landscapes. *arXiv preprint arXiv:1311.0376*, 2013.

C. Chen and M. Kerber. Persistent homology computation with a twist. In *Proc. of the 27th European Workshop on Computational Geometry*, 2011.

K.-S. Chou and X.-P. Zhu. *The curve shortening problem*. CRC Press, 2001.

N. Cristianini and J. Shawe-Taylor. An introduction to support vector machines and other kernel-based learning methods. Cambridge university press, 2000.

V. de Silva and R. Ghrist. Coordinate-free coverage in sensor networks with controlled boundaries via homology. *The International Journal of Robotics Research*, 25(12):1205–1222, 2006.

V. de Silva and R. Ghrist. Coverage in sensor networks via persistent homology. *Algebraic & Geometric Topology*, 7(1):339–358, 2007.

M. P. Deisenroth, D. Fox, and C. E. Rasmussen. Gaussian processes for data-efficient learning in robotics and control. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 99 (PrePrints):1, 2013. ISSN 0162-8828. .

R. Diankov and J. Kuffner. OpenRAVE: A Planning Architecture for Autonomous Robotics. Technical Report CMU-RI-TR-08-34, Robotics Institute, Pittsburgh, PA, July 2008.

R. A. Dwyer. Higher-dimensional voronoi diagrams in linear expected time. *Discrete & Computational Geometry*, 6(1): 343–367, 1991.

H. Edelsbrunner. Weighted alpha shapes. 1992.

H. Edelsbrunner. The union of balls and its dual shape. *Discrete and Comp. Geometry*, 13(1):415–440, 1995.

H. Edelsbrunner and J. Harer. Persistent homology-a survey. *Contemporary mathematics*, 453:257–282, 2008.

H. Edelsbrunner and J. L. Harer. Computational topology: an introduction. AMS Bookstore, 2010.

H. Edelsbrunner and E. P. Mücke. Three-dimensional alpha shapes. *ACM Transactions on Graphics (TOG)*, 13(1):43–72, 1994.

H. Edelsbrunner, D. Letscher, and A. Zomorodian. Topological persistence and simplification. *Discrete and Computational Geometry*, 28(4):511–533, 2002.

B. T. Fasy, F. Lecci, A. Rinaldo, L. Wasserman, S. Balakrishnan, and A. Singh. Statistical inference for persistent homology: Confidence sets for persistence diagrams. *arXiv preprint arXiv:1303.7117*, 2013.

A. Hatcher. Algebraic topology. Cambridge University Press, Cambridge, 2002.

L. Jaillet and T. Siméon. Path deformation roadmaps: Compact graphs with useful cycles for motion planning. *Int. Journal of Robotics Research*, 27(11-12):1175–1188, 2008a.

L. Jaillet and T. Siméon. Path deformation roadmaps. In S. Akella, N. M. Amato, W. H. Huang, and B. Mishra, editors, *Algorithmic Foundation of Robotics VII*, volume 47 of *Springer Tracts in Advanced Robotics*, pages 19–34. Springer, 2008b.

L. Jaillet, J. Cortés, and T. Siméon. Sampling-Based Path Planning on Configuration-Space Costmaps. *IEEE Transactions on Robotics*, 26:635–646, 2010. .

S. Karaman and E. Frazzoli. Sampling-based algorithms for optimal motion planning. *Int. Journal of Robotics Research*, 30(7):846–894, 2011.

P. Katsikouli, R. Sarkar, and J. Gao. Persistence based online signal and trajectory simplification for mobile devices. In *International Conference on Advances in Geographic Information Systems (ACM SIGSPATIAL)*. 2014.

L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Trans. on Robotics and Automation*, 12(4):566–580, 1996.

S. Kim, K. Sreenath, S. Bhattacharya, and V. Kumar. Optimal trajectory generation under homology class constraints. In *51st IEEE Conf. on Decision and Control*, 10-13 Dec 2012.

R. A. Knepper, S. S. Srinivasa, and M. T. Mason. Toward a deeper understanding of motion alternatives via an equivalence relation on local paths. *Int. Journal of Robotics Research*, 31(2):167–186, 2012.

J.-C. Latombe. Robot motion planning. Springer, 1991.

S. M. LaValle. Planning algorithms. Cambridge University Press, 2006.

S. M. LaValle and J. J. Kuffner. Rapidly-Exploring Random Trees: Progress and Prospects. In B. R. Donald, K. M. Lynch, and D. Rus, editors, *Algorithmic and Computational Robotics: New Directions*, pages 293–308, Wellesley, MA, 2001. A K Peters.

S. R. Lindemann and S. M. LaValle. Current issues in sampling-based motion planning. In *Robotics Research*, pages 36–54. Springer, 2005.

B. Luders, M. Kothari, and J. How. Chance constrained rrt for probabilistic robustness to environmental uncertainty. In *AIAA Guidance, Navigation, and Control Conference*. American Institute of Aeronautics and Astronautics, 2014/11/07 2010. .

J. Mainprice, E. A. Sisbot, L. Jaillet, J. Cortés, R. Alami, and T. Siméon. Planning human-aware motions using a sampling-based costmap planner. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 5012–5017. IEEE, 2011.

E. Masehian and D. Sedighizadeh. Classic and heuristic approaches in robot motion planning - a chronological review. *World Academy of Science, Engineering and Technology*, 23:101–106, 2007.

P. McMullen. The maximum numbers of faces of a convex polytope. *Mathematika*, 17(02):179–184, 1970.

B. Morris and M. Trivedi. Learning trajectory patterns by clustering: Experimental studies and comparative evaluation. In *IEEE Int. Conf. on Comp. Vision and Pattern Recognition (CVPR'09)*, pages 312–319. IEEE, 2009.

J. R. Munkres. Elements of algebraic topology. volume 2. Addison-Wesley Reading, 1984.

P. Niyogi, S. Smale, and S. Weinberger. Finding the homology of submanifolds with high confidence from random samples. *Discrete and Comp. Geometry*, 39(1-3):419–441, 2008.

F. T. Pokorny, C. H. Ek, H. Kjellström, and D. Kragic. Topological constraints and kernel-based density estimation. In *Advances in Neural Information Processing Systems 25, Workshop on Algebraic Topology and Machine Learning*. 2012.

F. T. Pokorny, M. Hawasly, and S. Ramamoorthy. Multiscale topological trajectory classification with persistent homology. In *Proceedings of Robotics: Science and Systems*, Berkeley, USA, July 2014.

C. E. Rasmussen and C. Williams. Gaussian processes for machine learning. In *Gaussian Processes for Machine Learning*. MIT Press, 2006.

J. T. Schwartz and M. Sharir. On the "piano movers" problem. II. General techniques for computing topological properties of real algebraic manifolds. *Advances in Applied Mathematics*, 4(3): 298–351, 1983.

E. Sisbot and R. Alami. A human-aware manipulation planner. *Robotics, IEEE Transactions on*, 28(5):1045–1057, Oct 2012. ISSN 1552-3098. .

L. Zhang, Y. J. Kim, and D. Manocha. A hybrid approach for complete motion planning. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, (IROS'07)*, pages 7–14. IEEE, 2007.

Y. Zheng and X. Zhou. Computing with spatial trajectories. Springer, 2011.

A. Zomorodian. Fast construction of the vietoris-rips complex. *Computers & Graphics*, 34(3):263–271, 2010.

Appendix

Here, we provide further details of the proof of Lemma 4.1. While we could not find the exact statement of this lemma in the literature, this result is a direct consequence of the abstract decomposition theorems of persistent homology (Edelsbrunner and Harer, 2010). We sketch the details of direct proofs which are obtained simply by reasoning about reduced matrices and which do not require further theory to provide a 'hands on' practical approach to these results. Let us recall that we denote by R the reduced boundary matrix, by D the original boundary matrix, and $R = DV$ for the transformation matrix V. When M is a matrix M_i denotes the i^{th} column and $low(M_i)$ is defined to be the index of the lowest non-zero element of M_i when $M_i \neq 0$ and low is undefined otherwise. We begin with two claims:

Claim 1. Let $(i,j) \in P = \{(i,j) : R_j \neq 0 \text{ and } i = low(R_j)\}$. Then $R_i = 0$.

Proof. Suppose $R_i \neq 0$ instead. Then $R_i \notin B(K_{i-1}) = span(R_1, \ldots, R_{i-1})$ since low is injective on the non-zero columns of R since R is reduced. We have $\partial R_j = \partial(\sigma_i + w) = \partial\sigma_i + \eta = \partial\partial V_j = 0$, for some $w \in C(K_{i-1})$, $\eta = \partial w \in B(K_{i-1})$ and σ_i denotes the i^{th} simplex in the filtration. Hence $\partial\sigma_i \in B(K_{i-1})$, but $R_i = \partial V_i = \partial\sigma_i + w'$ for some $w' \in B(K_{i-1})$. But then $R_i \in B(K_{i-1})$. A contradiction. □

Claim 2. With the same notation as in the main lemma, so $K = \cup_{i=1}^{n}\sigma_i$, we have

a) $\#P = \#$non-zero columns of R, where $(i,j) \in P$ corresponds to column R_j.
b) $\#P + \#E = \#$zero columns of R, where $(i,j) \in P$ corresponds to column R_i and $s \in E$ to column R_s.
c) $2\#P + \#E = n$

Proof. Part a): If $R_i \neq 0$, $low(R_i) = s$ for some s, so $(s,i) \in P$ and s is unique since R is reduced. Also, for each $(s,i) \in P$, $R_i \neq 0$. Part b): If $R_i = 0$, either there exists a j such that $i = low(R_j)$ in which case $(i,j) \in P$ and this j is unique since R is reduced, or there is no such j, in which case $i \in E$ by definition of E. Part c) follows from the above since each column of R is either zero or non-zero and R has n columns. □

Lemma. Let $K_1 \subset \ldots \subset K_n$ be a simplexwise filtration of simplicial complexes, let $R = DV$ denote the reduced boundary matrix after applying the left-to-right reduction algorithm, and let $E_p \subseteq E$, $P_p \subseteq P$ denote those elements corresponding to p-cycles only. For $1 \leqslant i \leqslant n$, a basis of $Z_p(K_i)$ is given by $S_p^i = \{R_t : (s,t) \in P_p, s \leqslant i\} \cup \{V_s : s \in E_p, s \leqslant i\}$, and, for $1 \leqslant i \leqslant j \leqslant n$, the image of the set

$$T_p^{i,j} = \{R_t : (s,t) \in P_p, s \leqslant i, t > j\} \cup \{V_s : s \in E_p, s \leqslant i\}$$

under the quotient map

$$q : Z_p(K_i) \twoheadrightarrow H_p^{i,j} = Z_p(K_i)/(B_p(K_j) \cap Z_p(K_i))$$

forms a basis of $H_p^{i,j}$. Finally $\#E_p = \dim(H_p(K_n))$.

Proof. Let us first show that S_p^i forms a basis for $Z_p(K_i)$.

Note that each column of of R only contains non-zero coefficients corresponding to simplices of a single fixed dimension each since column additions are only performed among columns of the same dimension. It is hence clear that $S_p^i \subseteq C_p(K_i)$. Note furthermore that $S_p^i \subseteq Z_p(K_i)$ since, for each R_t corresponding to $(s,t) \in P_p$, $s \leqslant i$, we have $\partial R_t = \partial\partial V_t = 0$. Similarly, for V_s, for $s \in E_p$ and $s \leqslant i$, we have $\partial V_s = R_s = 0$ by the definition of E_p. Next, we show that the elements of S_p^i are linearly independent. Define $A_p^i = \{R_t : (s,t) \in P_p, s \leqslant i\}$ and $B_p^i = \{V_s : s \in E_p, s \leqslant i\}$. We observe that low is injective on A_p^i. To see this, assume that $(i,j), (i,j') \in P_p$ for $j \neq j'$. Without loss of generality, assume $j < j'$, then we obtain a contradiction since R would not be reduced as we could add R_j to R_j' to further reduce R. Similarly, $low(V_i) = i$ for all $i \leqslant n$, as can be seen by observing that V is the result of a sequence of left-to-right column additions performed on the identity matrix, so low is also injective on B_p^i. Finally $low(A_p^i) \cap low(B_p^i) = \emptyset$ by the definition of E and P. Hence low is injective on $S_p^i = A_p^i \cup B_p^i$ and the set is hence linearly independent. To show that S_p^i is a basis for $Z_p(K_i)$, we now just need to show $\dim(Z_p(K_i) = \#S_p^i$.

Since the upper-left submatrices of the reduced matrix R which correspond to the subcomplexes K_i for $i \leqslant n$ are themselves reduced, and due to the grading by p, it is not difficult to extend the reasoning in Claim 2 to show that

$\dim B_p(K_i) = \#P_p^{\leqslant i}$ and $\dim C_p(K_i) = 2\#P_p^{\leqslant i} + \#E_p^{\leqslant i}$, for all $i \leqslant n$, where we define $P_p^{\leqslant i} = \{(a,j) : R_j \neq 0, a = low(R_j), \dim(\sigma_a) = p \text{ and } j \leqslant i\}$ and $E_p^{\leqslant i} = \{t : R_t = 0, low(R_j) \neq t \text{ for all } j \in \{1,\ldots,i\} \text{ and } \dim(\sigma_t) = p\}$. By the rank-nullity theorem, it follows that $\dim(Z_p(K_i)) = \#P_p^{\leqslant i} + \#E_p^{\leqslant i}$ for all $i \leqslant n$. Since it is clear that this is also the cardinality of the set S_p^i, the result follows.

Next, we prove the statement about $T_p^{i,j}$. Note that $T_p^{i,j}$ is a linearly independent set since low is injective on $T_p^{i,j}$. We need to show that $B_p(K_j) \cap Z_p(K_i)$, $i \leqslant j$, is spanned by $S_p^i - T_p^{i,j} = \{R_t : (s,t) \in P_p, s \leqslant i \text{ and } t \leqslant j\}$. First, observe that

$$B_p(K_j) = span\{R_t : (s,t) \in P_p, t \leqslant j\}.$$

This follows since $span(V_1,\ldots,V_j) = C(K_j)$ and if $\partial V_i \neq 0$, then $R_i = \partial V_i$ is in the above spanning set which is also linearly independent.

Next, we know that $Z_p(K_i)$ has basis S_p^i. From this, we observe, $S_p^i - T_p^{i,j} \subset B_p(K_j) \cap Z_p(K_i)$. Finally, $B_p(K_j) + Z_p(K_i)$ has basis $\{R_t : (s,t) \in P_p, t \leqslant j\} \cup S_p^i = \{V_s : s \in E_p, s \leqslant i\} \cup \{R_t : (s,t) \in P_p, t \leqslant j \text{ or } s \leqslant i\}$ which can be written as a disjoint union of S_p^i and $\{R_t : (s,t) \in P_p, i < s \leqslant t \leqslant j\}$. Counting dimensions, we find that $\dim(B_p(K_j) \cap Z_p(K_i)) = \dim(B_p(K_j)) + \dim(Z_p(K_i)) - \dim(B_p(K_j) + Z_p(K_i)) = \#\{R_t : (s,t) \in P_p, t \leqslant j\} - \#\{R_t : (s,t) \in P_p : i < s \leqslant t \leqslant j\} = \#\{R_t : (s,t) \in P_p : s \leqslant i \text{ and } t \leqslant j\} = \#(S_p^i - T_p^{i,j})$. To see that $\#E_p = \dim(H_p(K_n))$, observe that $H_p(K_n) = H_p^{n,n}(K_n)$ and $T_p^{n,n} = E_p$. □